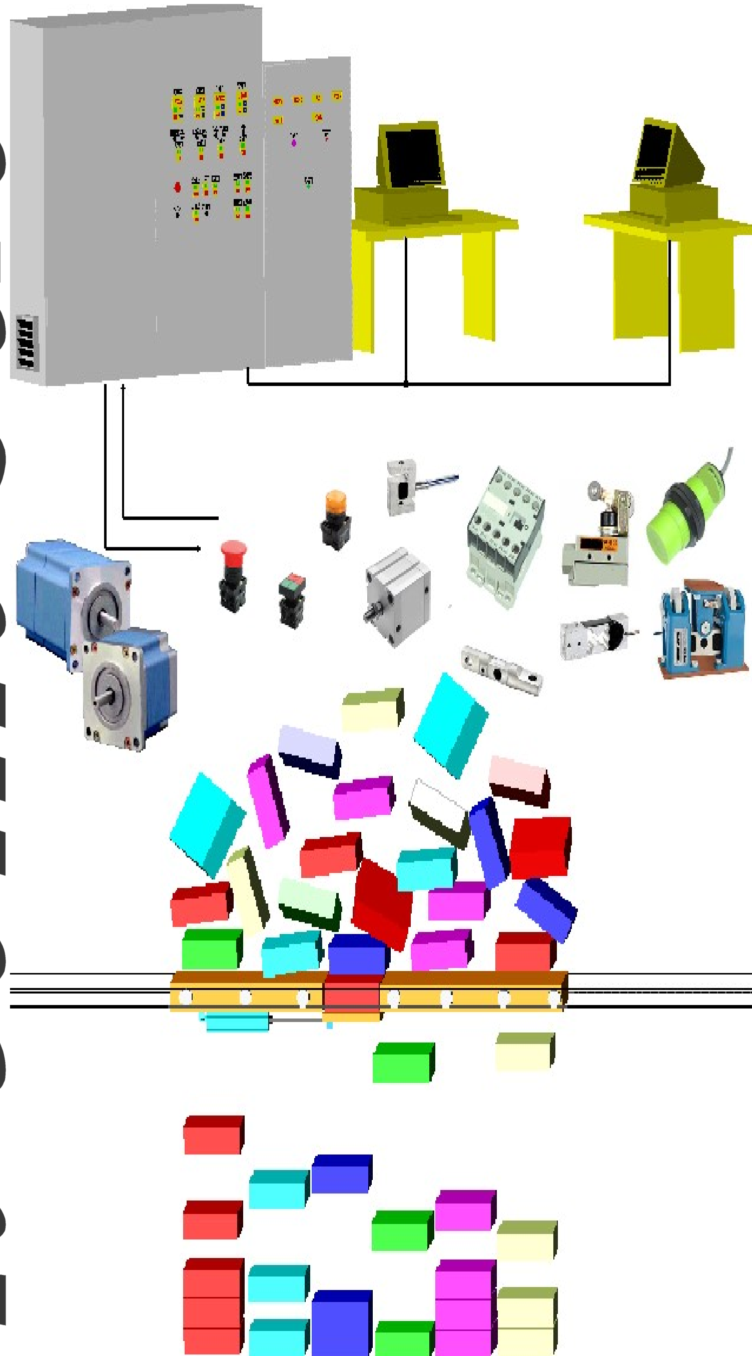
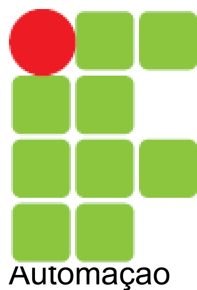
Automação



Prof. Eng. Giovani Batista de Souza



INSTITUTO FEDERAL
SANTA CATARINA
Campus Araranguá

Fevereiro- 2010



MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO CIÊNCIA E TECNOLOGIA DE SANTA CATARINA
CAMPUS DE ARARANGUÁ

Apostila de automação industrial desenvolvida a partir de um trabalho de Automação 1 e 2 intitulado “Automação com controladores lógicos” do próprio autor.

Histórico versões:

- Versão 1 – Fevereiro 2010 - Prof. Giovani Batista de Souza

“A escola é um espaço para os alunos aprenderem, não para os professores ensinarem.”

Berthold

Sumário

“Ainda que eu falasse

As línguas dos homens e dos anjos,

E não tivesse AMOR, seria como

O metal que soa ou

Como o sino que toca

E ainda que tivesse o dom de profecia,

E conhecesse todos os mistérios

E toda a ciência,

E ainda que tivesse toda a fé,

De maneira tal que

Movimentasse montanhas,

E não tivesse AMOR,

Nada seria.”

(Cartas de São Paulo Apostolo aos Coríntios I,
Capítulo 13, versículos 1 e 2, Bíblia Católica)

1 - Sistemas Automáticos

> *"... a ciência constrói robôs que... calculam a uma velocidade vertinosa e que são capazes de vencer o poder, criando uma grande devastação. Infelizmente, a ciência ainda não conseguiu produzir o homem que pensa não por meio de fios e engrenagens, mas com seu ser sensível, e capaz de determinar o destino dos robôs. Esse ser sensível é que devemos educar, não apenas para criar e animar os robôs, mas para dominá-los e exaltar os elementos de consciência e humanidade que são a grandeza e a razão de ser do Homem."*
>
> FREINET (*apud* PUEBLA, 1997, p.20).

1.1 Introdução

A automação é um processo de engenharia que identifica comportamentos repetitivos em máquinas e processos, a fim de determinar a utilização de mecanismos e sensores para reprodução destes movimentos e ações. Garantindo pelo menos um dos seguintes objetivos:

- Aumento da Produtividade,
- Aumento da Qualidade,
- Aumento da Segurança e
- Evitar contato manual.

Em um sistema automático a produtividade é normalmente maior, pois a sincronia dos mecanismos é feita de tal modo a reduzir os intervalos de tempos entre duas tarefas.

1. Exemplo1: Sincronia das esmaltadeiras em linha de esmaltação cerâmica.
2. Exemplo2: Aumento de produção em uma máquina de envase automático de refrigerantes.
3. Exemplo3: Sistema de dobra e furação automático, reduz o tempo de produção, pois evita medição de cada operação.

A qualidade do produto é assegurada, pois os movimentos dos mecanismos são precisos, garantindo a repetibilidade dos movimentos indefinidamente e a produção de produtos sempre com a mesma qualidade.

- Exemplo1: Sistema de dosagem de ingredientes, pois permite que a quantidade de cada produto dosado seja sempre a mesma.
- Exemplo2: Aumento da qualidade de queima em um forno cerâmico para evitar a produção de produtos de classificação de 2ª ou 3ª.
- Exemplo3: Linha para corte com precisão de chapa metálica evitando a sobra de cavacos.

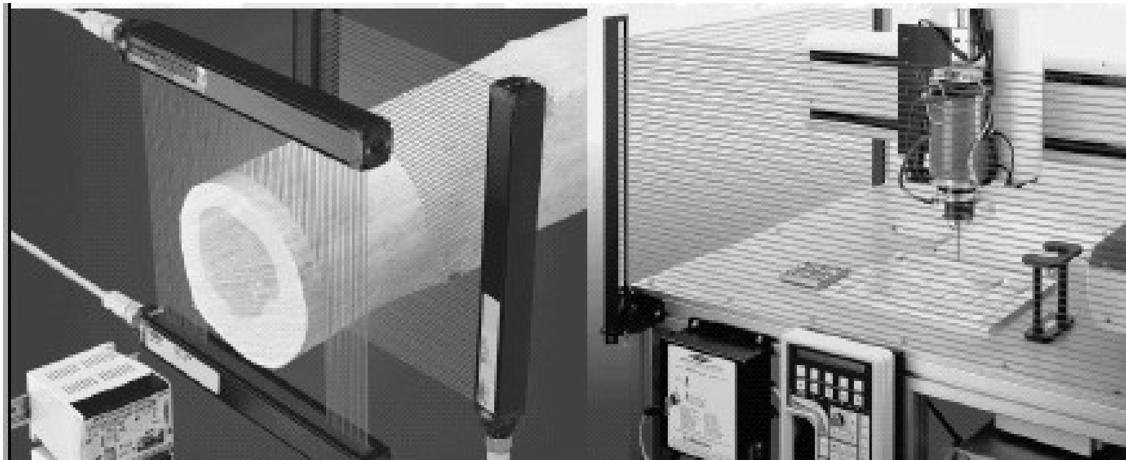


Figura 1-1 Sensores de barreiras

Alguns mecanismos são feitos também para impedir o contato humano, evitando acidentes de trabalho, desta forma contribuindo com a segurança do usuário. E também na higienização de produtos ou embalagens para produto de consumo humano.

4. Exemplo 1: Manipulação de produtos químicos.
5. Exemplo 2: Produção de medicamentos.
6. Exemplo 3: Produção de pratos e copos descartáveis.
7. Exemplo 4: Envase de produtos alimentícios.

Existem também automações que são feitas apenas para controlar alarmes e falhas de um sistema, contribuindo com a segurança e evitando expor o ser humano à ambiente potencialmente perigoso.

- Exemplo 1: Controle de alarmes em planta de extração de óleo de soja.
- Exemplo 2: Sistema de falhas em plataforma de petróleo.
- Exemplo 3: Sistema de falhas e barreiras de proteção para prensas.
- Exemplo 4: Controle de Temperatura de um autoforno.

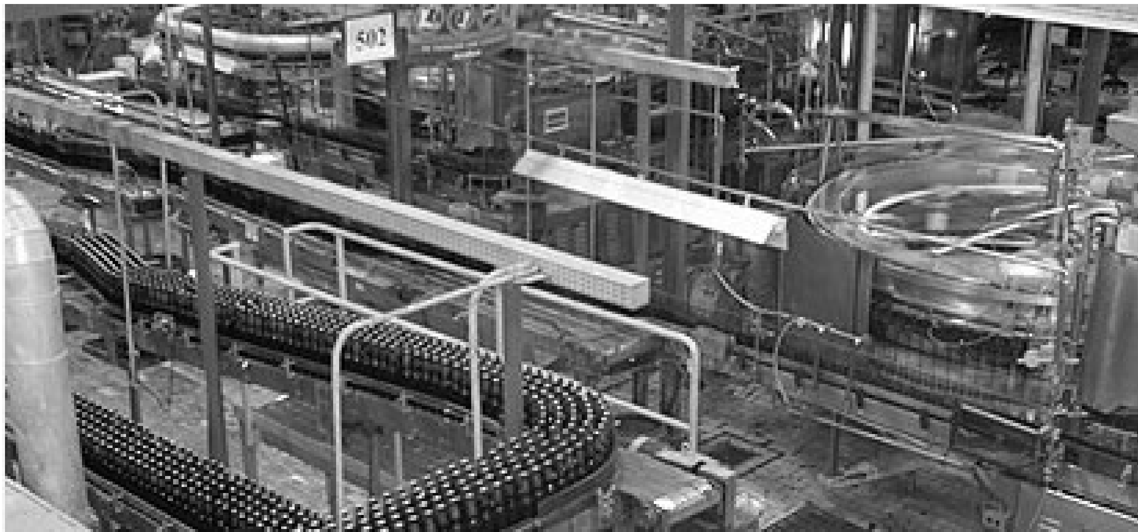


Figura 1-2 Vista interna de uma fábrica

A automação também é importante para segurança da máquina, impedindo que os mecanismos da máquina sejam autodestruídos durante o funcionamento da mesma.

8. Exemplo1: Controle de alimentador de prensa hidráulica.
9. Exemplo2: Controle de pressão de sistema de filtragem.
10. Exemplo3: Controle de parada a vazio para elevadores de grãos e pós.

O processo de automação é dividido em duas partes distintas: automatização ou mecanização e automação propriamente dita. A

automatização ou mecanização está diretamente ligada aos atuadores e mecanismos capazes de efetuar os movimentos de repetição identificados. Nesta etapa é dividida em parte elétrica e mecânica, tais como: atuadores pneumáticos e eletromagnéticos, motores elétricos, servomotores, etc. A automatização propriamente dita, corresponde à atividade de desenvolvimento de integração dos movimentos de todos os mecanismos de forma ordenada e pré-estabelecida. Este método de automatização foi o primeiro método de seqüenciação a ser desenvolvido e era bastante utilizado, mas não permitia a automação total de processos muito complexos.

O método de controle de automação é o que utiliza sinais elétricos para acionamento dos movimentos, através de “máquinas processadoras de dados”. Este é o método mais utilizado hoje em dia devido ao menor número de manutenção, a flexibilidade e ao custo final de implantação. Com a evolução da eletrônica, permitiu-se a construção de equipamentos eletrônicos extremamente pequenos, confiáveis e capazes de executar uma enormidade de operações matemáticas, o que possibilitou o desenvolvido dos Controladores Lógicos a preços bastante competitivos. Além do preço um sistema automático com controlador lógico, tem um menor número de manutenção, devido ao menor número de partes móveis. Também tem uma flexibilidade maior, pois permite alterar o sequenciamento da máquina trocando apenas o seu programa.

Outra facilidade da utilização de controladores lógicos é a utilização de programas de monitoração via microcomputadores, denominados supervisórios. Estes programas permitem o controle remoto e o armazenamento de dados de produção, facilitando o gerenciamento através de relatórios e planilhas de produção e eventos do sistema.

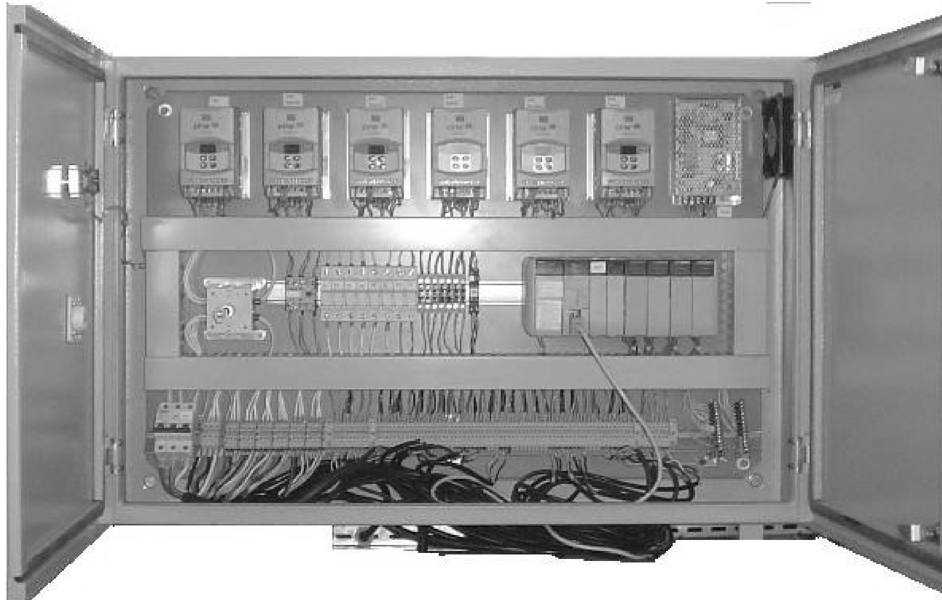


Figura 1-3 Vista interna de um painel elétrico



Figura 1-4 Vista externa de painel elétrico de força e comando

1.2 Histórico

O desejo de facilitar as atividades humanas começou com o surgimento das primeiras comunidades, com a utilização primeiramente da lança e depois de instrumentos agrários tais como pás e enxadas etc. Já o processo de automação de máquinas começa ainda na época da revolução industrial com a invenção de máquinas para produção de bens de consumo, para a indústria têxtil, o que acarretou um grande impulso da ciência que passou a aprimorar os processos, as máquinas e as fontes de energia.

Inicialmente a fonte de energia de propulsão das indústrias era a hidráulica ou eólica. A energia hidráulica funcionava através do movimento de uma enorme roda que girava devido à força do fluxo de um canal de água desviado para passar pelos canecos (aletas) que formavam a roda d'água. A rotação desta roda era responsável pelo movimento dos mecanismos da fábrica.



Figura 1-5 Sistema motriz antigo (Roda d'água)

A energia eólica era obtida de cataventos que movimentavam um eixo central sendo utilizado principalmente para bombeamento de água e para moer grãos.

Apresentamos um pequeno resumo da evolução da tecnologia que contribuíram para o desenvolvimento da automação industrial:

- ➲ Navegação com energia dos ventos, à vela pelos Fenícios (1000 A.C.);
- ➲ Moinhos de ventos já eram utilizados antes do século X;
- ➲ Invenção máquina de fiar (1764);
- ➲ Tear Hidráulico (1768);
- ➲ Máquina de Vapor (1769);
- ➲ Bateria Elétrica (1800) Alessandro Volta;
- ➲ Navegação com energia a vapor (1807);
- ➲ Relação entre eletricidade e magnetismo (1820), por Hans Christian Orsted;
- ➲ Eletromagnetismo (1864) por James Clerk Maxwell;
- ➲ Equação da força eletromotriz por Faraday (1831) e Lenz (1834);
- ➲ Rede elétrica pública em Londres (1882);

- ➲ Motor Elétrico CC (1886) através do gerador de CC por Werner Von Siemens;
- ➲ Motor Elétrico CA (1889) o Eng. Dobrowolsky patenteou este motor.

Neste trabalho focaremos o estudo de controle automático via controladores lógicos, que são circuitos eletrônicos, surgidos graças à evolução desta ciência. Esta evolução é bastante recente e ocorreu de forma muito rápida, para enfatizar esta afirmação é que colocamos abaixo um pequeno histórico da evolução da eletrônica em nossa sociedade:

- ➲ Transistor (1948) por Bradeen e Bradain;
- ➲ Demonstração prática da máquina CNC (1953);
- ➲ Circuito Integrado de Silício (1958) Jack Kilby;
- ➲ Padronização de comunicação serial RS232 (1960);
- ➲ Microprocessador (1971) Laboratório Intel;
- ➲ Aparecimento dos Controladores Lógicos na indústria automotiva (década 70);
- ➲ Mini-calculadoras eletrônicas (década de 70)
- ➲ Computador pessoal (1975) Edward Roberts;
- ➲ Máquina CNC com microprocessador (1977);
- ➲ Protocolo MODBUS (1979);
- ➲ PC – Padrão IBM (*Personal Computer*) (1981);
- ➲ Protocolo Profibus (1987);
- ➲ Normas para programação IEC 61131 (1992);
- ➲ Internet aberta e acessível a todos (1995).

1.3 Descrição de Sistemas Automáticos

Um sistema automático ou sistema de controle é dividido em duas partes tais como controle e sistema controlado.

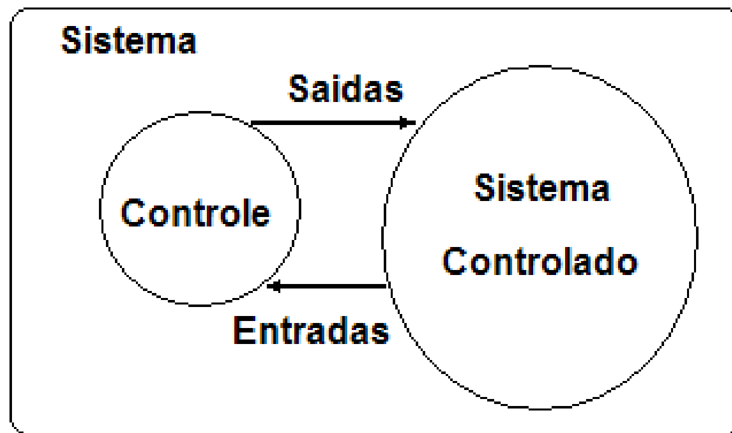


Figura 1-6 Sistema de controle

O controle é formado pelos equipamentos físicos que comandam todo o sistema, enviando ordens para a máquina controlada.

O sistema controlado corresponde à máquina e/ou processo final, que recebe as ordens e envia as informações necessárias para um perfeito controle do sistema.

A automação com sucesso é desenvolvida com a descrição detalhada de todas as partes de um sistema de controle, a fim de eliminar e/ou reduzir todos os erros de programação durante a posta em marcha do sistema.

2 - Sinais e Sensores

> "A dinâmica do ato de verificar encerra-se com a obtenção do dado ou informação que se busca, isto é, 'vê-se' ou 'não se vê' alguma coisa. E... pronto! Por si, a verificação não implica que o sujeito retire dela conseqüências novas e significativas."
>
> *LUCKESI, 1995, p.92*

2.1 Sinais e Sistemas

Sinal é uma descrição quantitativa de um determinado fenômeno, associado a um dado do meio. Exemplos de sinais: sonoros, visuais e elétricos.

O sistema é uma parte do meio que cria sinais próprios e que permite que ele se relacione com o meio ambiente. Exemplos de sistemas: circuitos elétricos, sistemas hidráulicos, sistemas mecânicos.

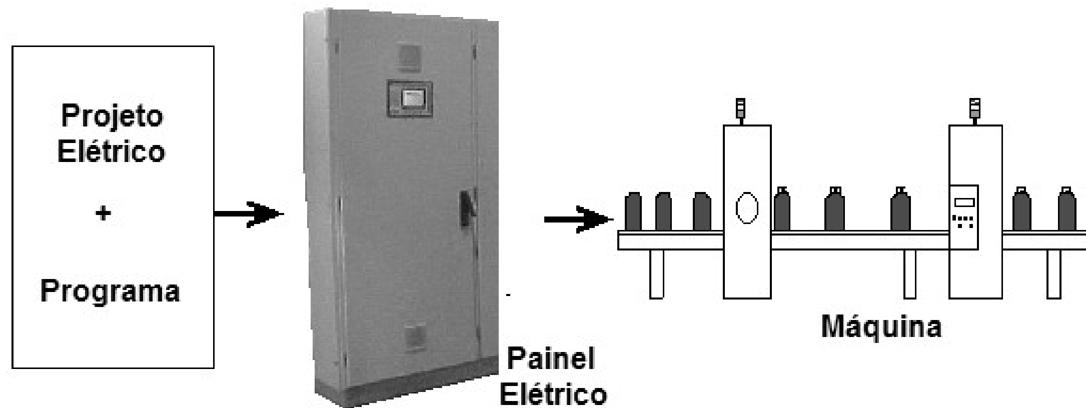


Figura 2-7 Sistema de controle

2.1.1 Entradas e saídas de um sistema

Todo sistema tem associado a ele um ou mais sinais. Os sinais que relacionam o sistema com o meio são designados como: sinais de entrada e sinais de saída. O meio atua sobre o sistema através dos sinais de entrada e o sistema atua sobre o meio através dos sinais de saída. A representação de um sistema em um meio genérico é feito da seguinte forma:

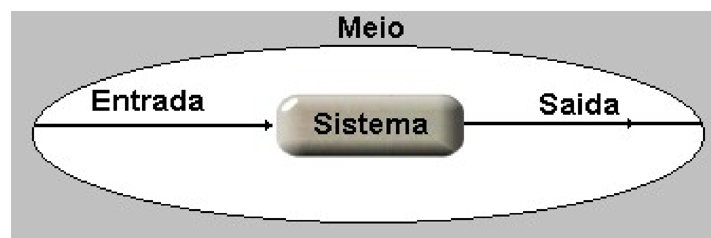


Figura 2-8 Representação sistema e meio

Exemplo: A Lâmpada elétrica recebe uma dada tensão (V) e emite luz e calor. Então o sistema lâmpada tem como sinal de entrada a tensão e como sinais de saída à luz e o calor.

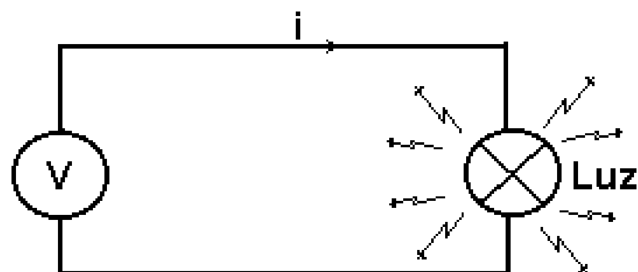


Figura 2-9 Exemplo sistema lâmpada

2.1.2 Sinais Contínuos e Discretos

O tempo é um fator muito importante nos sistemas de controle e automação, pois os sinais relacionados aos sistemas são funções do tempo. Em função do tempo os sinais são analisados de forma discreta ou contínua.

Sinais discretos são sinais definidos em um número contável de pontos no eixo dos tempos. Na figura 2-4 observamos que entre os pontos cinco e seis do eixo do tempo não temos o valor real do sinal.

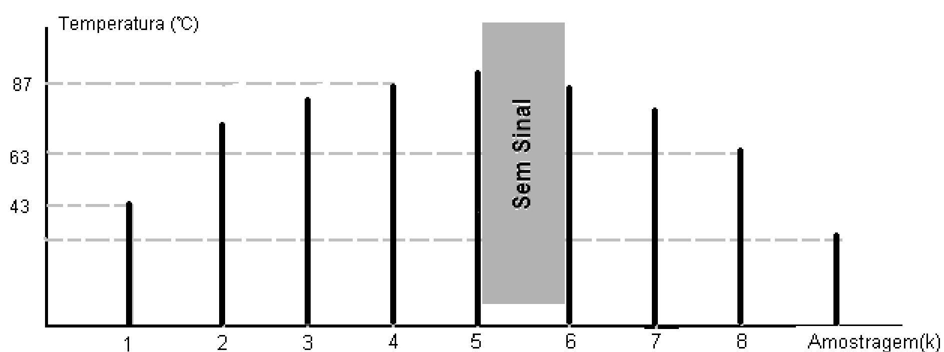


Figura 2-10 Sinais discretos

Sinais contínuos são sinais definidos no eixo do tempo em intervalos com infinitos pontos.

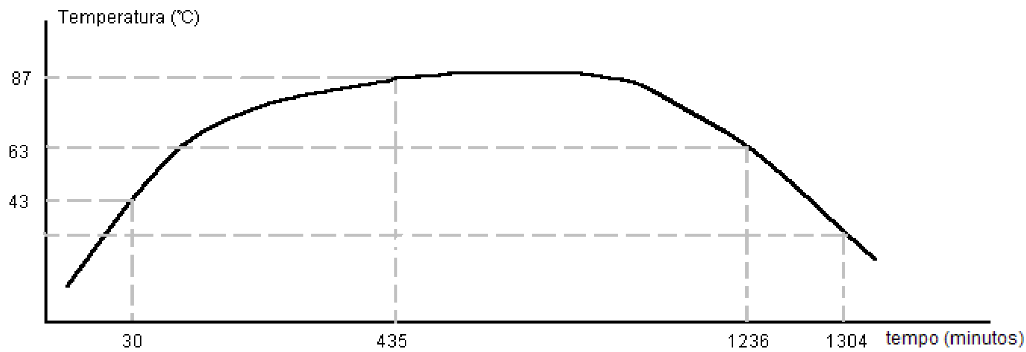


Figura 2-11 Temperatura ao longo de um dia em uma estufa

Geralmente trabalhamos com uma combinação de sinais contínuos e discretos, pois os conversores analógicos para digital (A/D) e digital para analógico (D/A) necessitam de tempo para conversão destas grandezas. Portanto para analisar matematicamente estas combinações é necessário então amostrar os sinais contínuos de forma a construir seqüências que as representam. A amostragem é feita através de coletas de informações do sinal contínuo em determinados instantes, chamadas amostras. O intervalo de coleta geralmente é constante e é chamado de período de tempo (T).

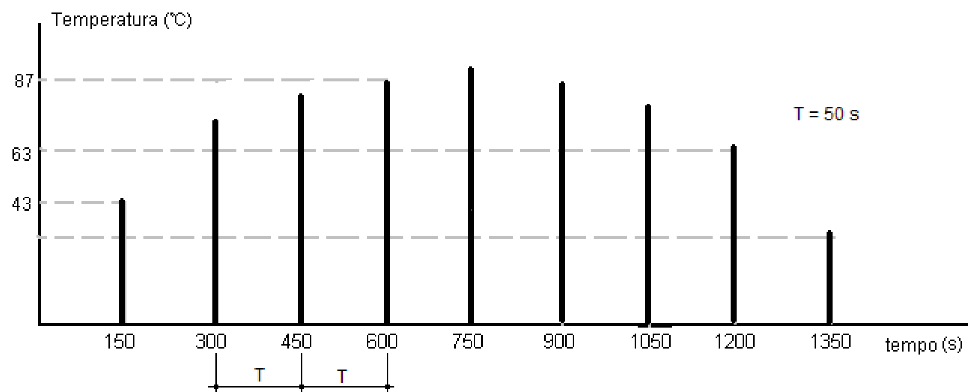


Figura 2-12 Amostragem de sinal ao longo de um período

2.2 Sinais Digitais

Sinais elétricos digitais são formados por sinais que tem seus níveis variando entre dois níveis de energia pré-estabelecidos, definidos como nível 0 (zero) e nível 1 (um). Representamos variáveis de nosso dia-a-dia através de sinais elétricos digitais, tais como:

- ➲ Aberto / fechado, através de sensor indutivo, magnético e fim de curso;
- ➲ Ligado / desligado através de contato auxiliar de um contator;
- ➲ Cheio e vazio através de sensor de nível tipo bóia ou capacitivo;
- ➲ Avançar / Retornar, comando para um cilindro pneumático;
- ➲ Posição Ocupada / Desocupada através de sensor ótico ou indutivo;
- ➲ Quente / Frio através de termostato;
- ➲ Pressão Baixa / Alta através de pressostato;
- ➲ Ligar e desligar, comando para um motor.

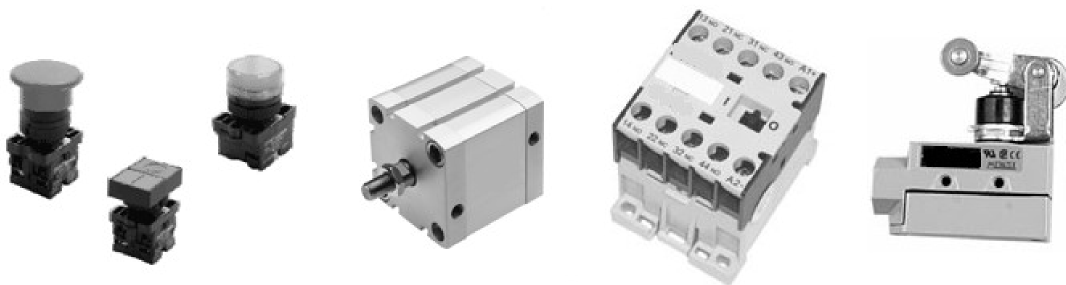


Figura 2-13 Dispositivos associados a sinais binários (Botoeira, cilindro, contator e fim de curso)

2.3 Sinais Analógicos

São sinais elétricos, continuo no eixo do tempo, que apresentam o nível da grandeza dentro de uma faixa de valor, podendo assumir qualquer valor entre os limites desta faixa. Idealmente um sinal analógico assume um valor entre os limites do infinito, de forma real estes valores terão uma faixa definida.

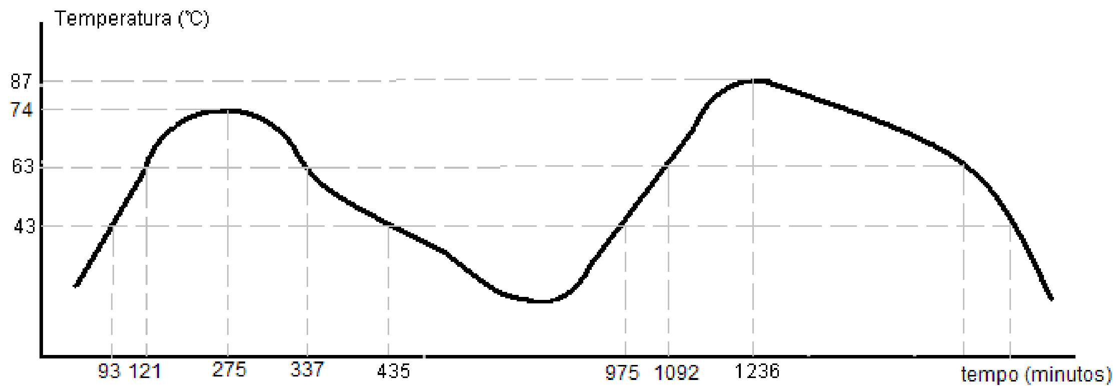


Figura 2-14 Curva variável analógica

As grandezas físicas relacionadas abaixo são utilizadas preferencialmente como grandezas analógicas, pois o controle das variáveis será feito de uma forma mais adequada.

- Temperatura;
- Pressão;
- Vazão;
- Nível de um tanque;
- Massa em uma balança;
- Luminosidade;
- Umidade;
- Velocidade;
- Posição;
- Tonalidade;
- Concentração de oxigênio;
- Nível pH.

2.4 Transmissores e Sensores

Os sensores são os elementos primários que transformam uma grandeza física qualquer em um sinal elétrico. E normalmente são utilizados em conjunto com equipamentos transmissores, que servem como amplificadores do sinal para transmissão à distância, bem como para conversão do sinal de entrada para um valor de tensão/corrente a níveis padronizados.

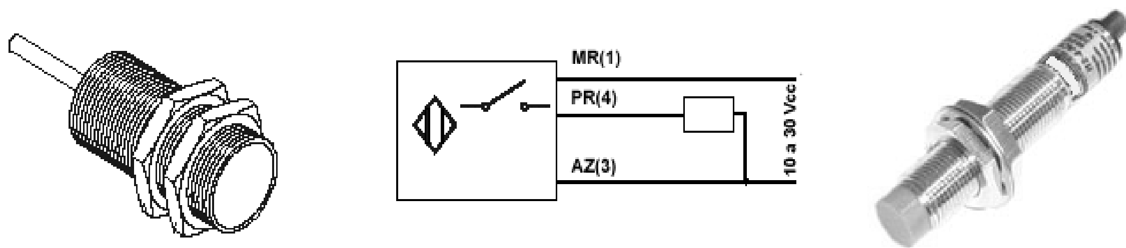


Figura 2-15 Sensor indutivo

Além dos sensores e transmissores, num sistema de coleta de sinais, também são utilizados os conversores, que são equipamentos eletrônicos que servem para converter um sinal primário em um sinal elétrico padronizado. Na figura 2-10 observamos a utilização de um conversor entre o sinal do sensor de temperatura e o controlador, neste caso o conversor foi utilizado para ajustar a faixa de valor obtida pelo sensor para a faixa de utilização do controlador (0 a 10 Vcc / 4 a 20 mA).

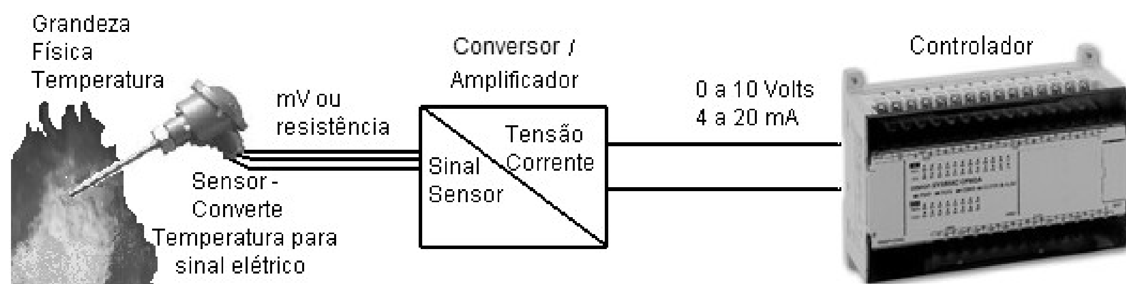


Figura 2-16 Sensor e transdutor

As entradas de contagens rápidas disponíveis na maioria dos controladores lógicos são alternativas a utilização de unidades de conversão analógicas para digital. Estas entradas são transformadas em entradas analógicas através da utilização de conversores das grandezas físicas para freqüência (0 a 10 kHz).

No exemplo indicado na figura 2-11, a temperatura obtida é inicialmente convertida em sinal de tensão/corrente padronizado e posteriormente este valor é transformado em um trem de pulso entre zero a 10 kHz. Utiliza-se esta opção para reduzir o custo de uma instalação e viabilizar um sistema de controle total com controlador, quando o mesmo não dispor de conversores analógicos para digital, ou o custo de seus módulos de expansões for exorbitante.

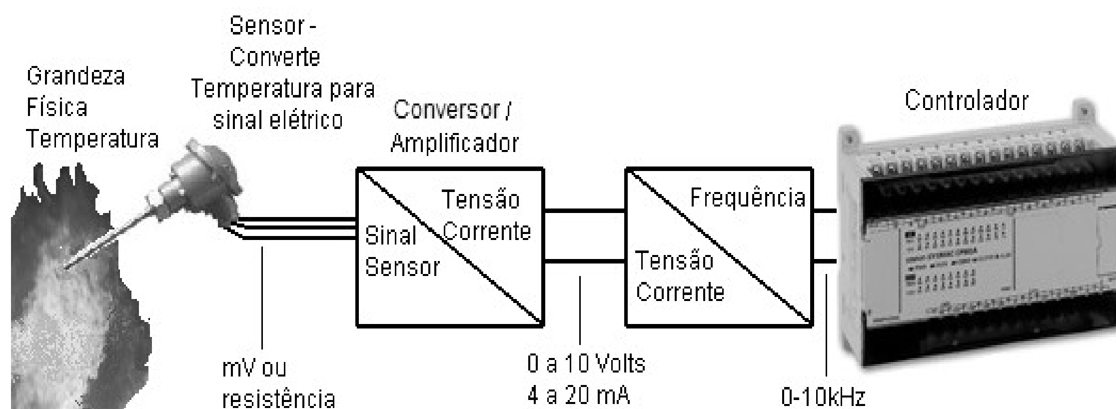


Figura 2-17 Leitura por contagem de pulsos

2.4.1 Transdutores de Temperaturas

Para medição de temperaturas existem vários tipos de sensores, o mais utilizado é o termopar. O princípio de funcionamento do termopar é a diferença de potencial que aparece no ponto de conexão entre dois metais diferentes, a qual é variável com a temperatura.

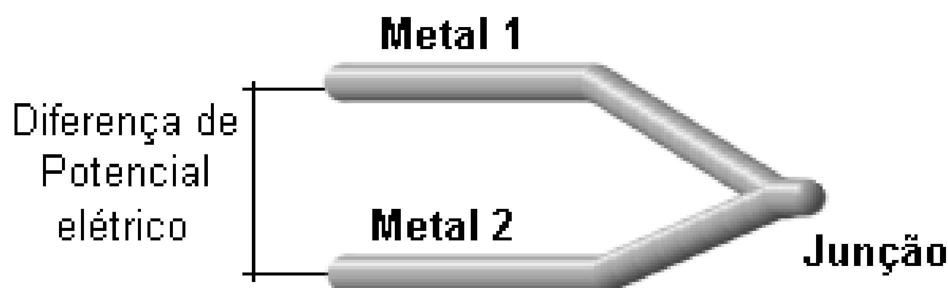


Figura 2-18 Termopar

A máxima temperatura de emprego dos termopares depende da constituição dos materiais que compõe a junta, e neste caso são padronizados como J, K, T, E, N, R, S e B conforme tabela 2.1.

Tipo	Liga (+)	Liga (-)	Limites de Temperaturas
J	Ferro	Constantan (Cobre-Niquel)	0 a 750 ºC

K	Cromel (Níquel – Cromo)	Alumel (Níquel-Alumínio)	0 a 1250 °C
T	Cobre	Constantan	-185 a 370 °C
E	Cromel	Constantan	0 a 870 °C
N	Nicrosil (Níquel-Cromo-Silicio)	Nisil (Níquel-Silicio-Magnésio)	-90 a 1300 °C
R	Cobre	Constantan	0 a 1480 °C
S	Cobre	Constantan	0 a 1480 °C
B	Platina Platina – 30% Rhadio	Platina Platina – 6% Rhadio	150 a 1820 °C

Tabela 2-1 Tipo de termopares

Os termopares são bastante utilizados devido ao pequeno tamanho e ampla faixa de utilização. Mas apresentam as seguintes desvantagens:

- A faixa de leitura não é linear;
- Precisam de conversores;
- Normalmente necessitam de cabos de compensação.

A curva tensão por temperatura de um termopar tipo J é apresentada na figura 2-13.

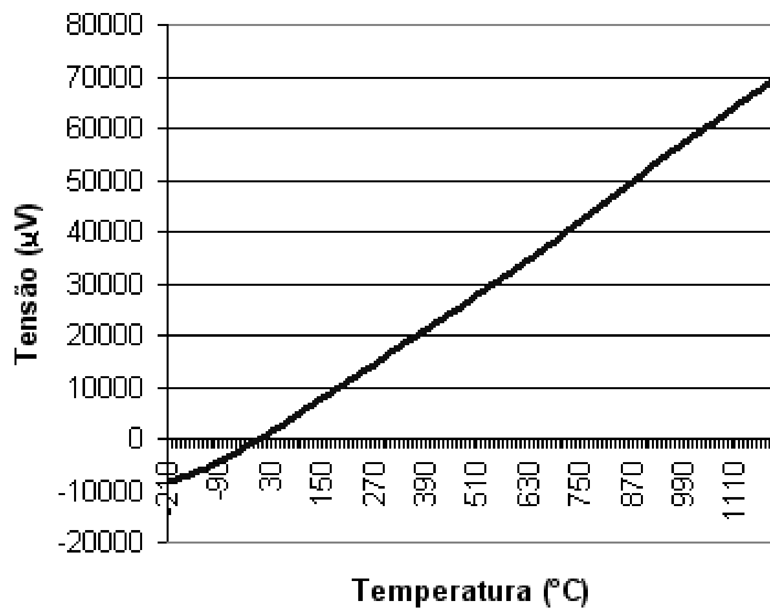


Figura 2-19 Relação Tensão x Temperatura do termopar tipo J

Outro método de medição de temperatura é a termoresistência, conhecida por PT100, que tem o principio de funcionamento fundamentado na dependência do valor da resistência elétrica de um condutor com a temperatura externa. A grande vantagem deste método é a alta precisão e boa sensibilidade. Para seu funcionamento faz-se necessário à utilização de uma fonte de energia constante. O nome PT100 vem do material utilizado platina (PT) e da resistência apresentada a zero grau Celsius (100 ohms). A faixa de leitura normal é de -220 a 500 ºC, para temperaturas mais elevadas, até 850 ºC, o sensor precisa passar por um tratamento químico

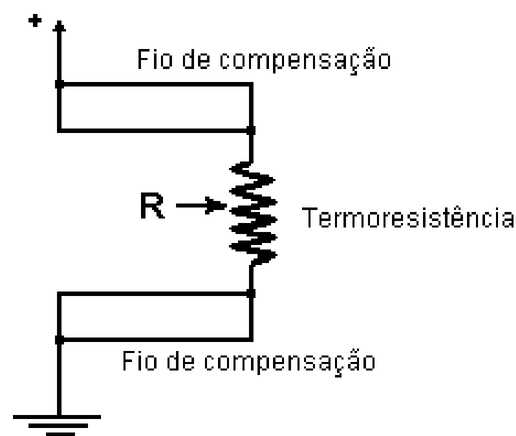


Figura 2-20 Representação de um PT100

Existe também para aplicações mais específicas o termistor, que são elementos semicondutores que variam a resistência dependendo da temperatura. Os termistores não realizam a conversão de forma linear e possuem baixa estabilidade a altas temperaturas.

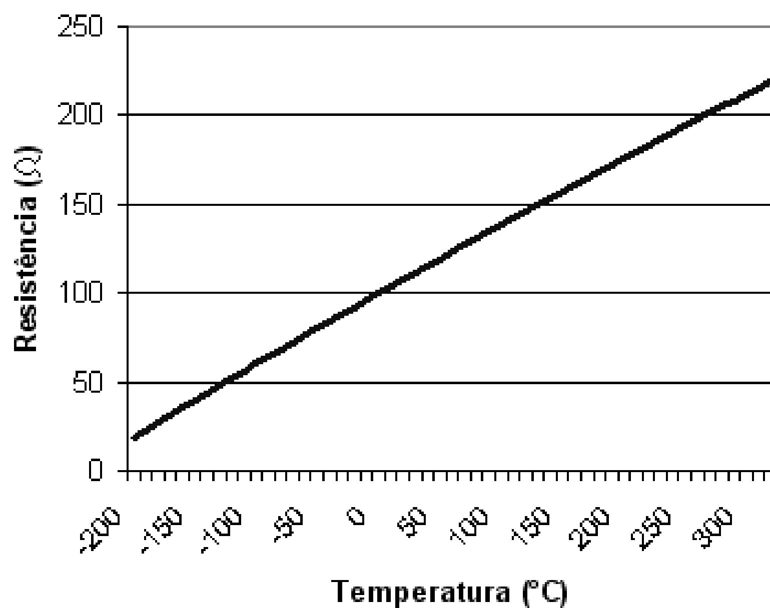


Figura 2-21 Relação tensão x resistência de um PT100

Como os sensores de temperaturas não disponibilizam diretamente o sinal físico medido em um sinal elétrico padrão de entradas analógicas (4 a 20

mA ou 0 a 10 V), normalmente são utilizados conversores de sinal para realizar esta transformação.

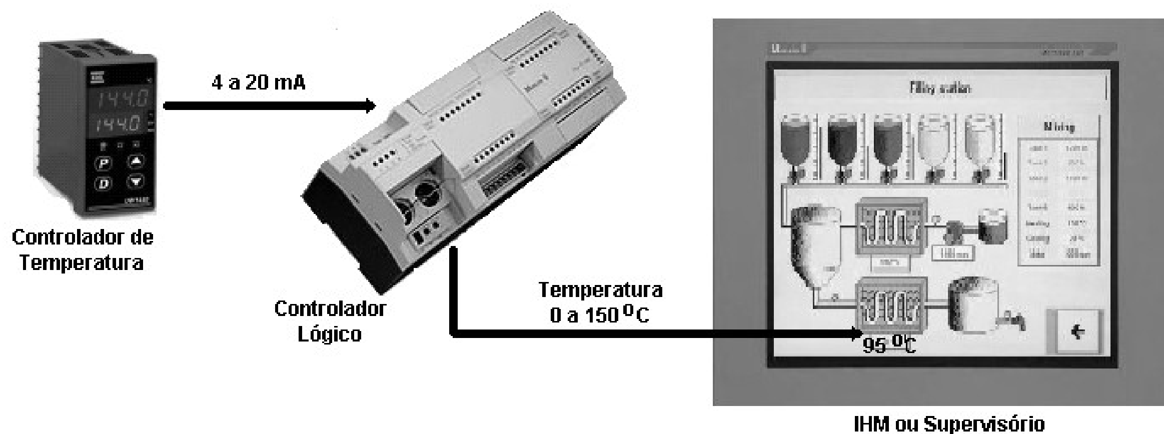


Figura 2-22 Representação de equipamentos para controle de temperatura

2.4.2 Transdutor de Pressão

Existem vários métodos para medição de pressão, o método mais utilizado é o da banda extensométrica ou *Strain Gauges*. Que tem seu funcionamento fundamentado no seguinte princípio: "*Um condutor elétrico ao ser submetido a um esforço mecânico, muda seu comprimento e seu diâmetro, portanto variando sua resistência elétrica*".

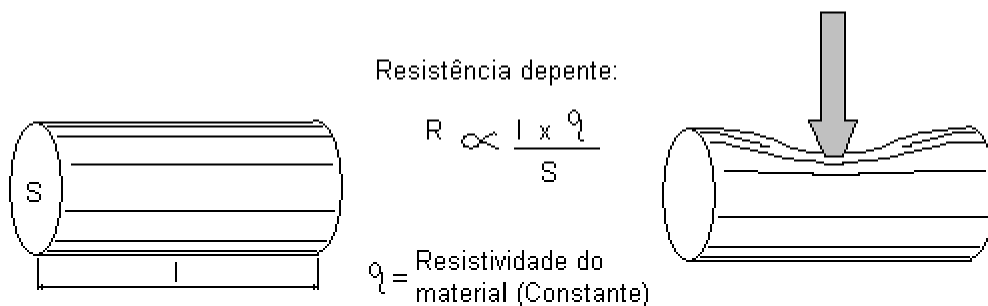


Figura 2-23 Influência pressão num condutor

Este método de medição de pressão também é utilizado para medição indireta de níveis de tanques, com a verificação da pressão na base do tanque. Este sensor é utilizado também como sensor mássico em balanças onde são denominados de células de carga.



Figura 2-24 Células de carga para balança

2.4.3 Transdutor de Nível

Existem vários métodos para a medição de nível e um dos mais modernos e precisos é o sensor ultra-sônico, que funciona através da leitura de tempo que um sinal sonoro leva para sair do medidor, refletir no nível do tanque e retornar ao medidor. O nível é calculado através da diferença da altura total do medidor em relação ao tanque com a distância projetada pelo som.

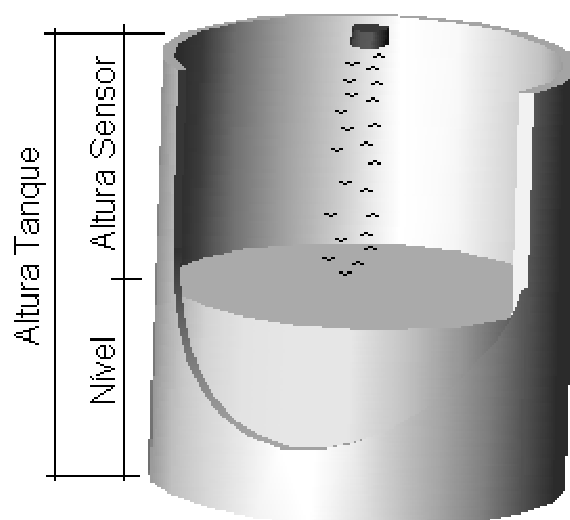


Figura 2-25 Aplicação sensor ultra-sônico

2.4.4 Transdutores de Posição

O sensor de posição é usado para determinar o espaço percorrido por um equipamento com objetivo de localização e posicionamento.

O encoder é o dispositivo utilizado para determinar a posição de um manipulador, existindo dois tipos de enconders, o absoluto e o incremental. A figura 2-20 mostra o aspecto físico externo de três encoder.



Figura 2-26 Encoder

O encoder incremental é um dispositivo elétrico que gera um determinado número de pulsos constantes por cada revolução de seu eixo. Para determinar a direção dos pulsos existem dois canais, denominados A e B, que geram pulsos defasados entre si de meio pulso. Existe também um terceiro sinal "Z" que gera um pulso a cada revolução do encoder, e serve como referência de rotação. A especificação do encoder é feita através do número de pulsos por volta e das características mecânicas de fixação e proteção. A figura 2-20 mostra a função de saída em função do tempo, para as saídas A, B e Z.

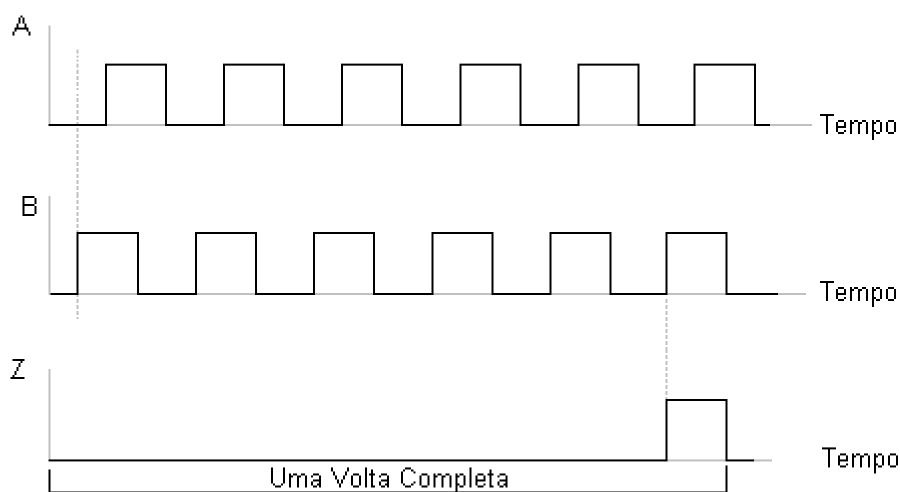


Figura 2-27 Encoder incremental – pulsos

A figura 2-21 demonstra a construção do disco interno e dos leitores óticos que permite gera o código do gráfico da figura 2-20.

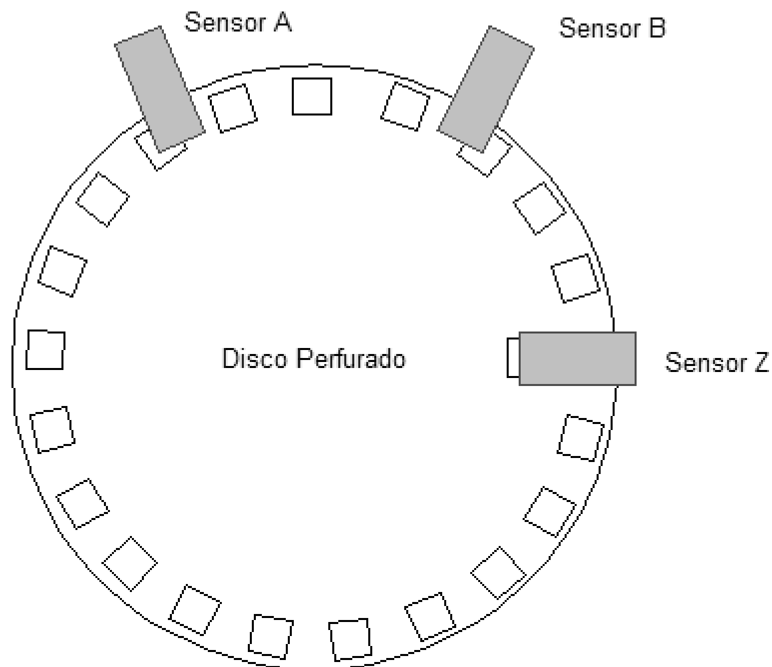


Figura 2-28 Disco do enconder incremental

O encoder absoluto gera um código binário por fração de volta em torno de seu eixo (figura 2-22), não perdendo seu posicionamento após a desenergização do sistema. O encoder absoluto possui uma furação diferente do encoder incremental e possui um sensor ótico para cada trilha sensora (figura 2-23).

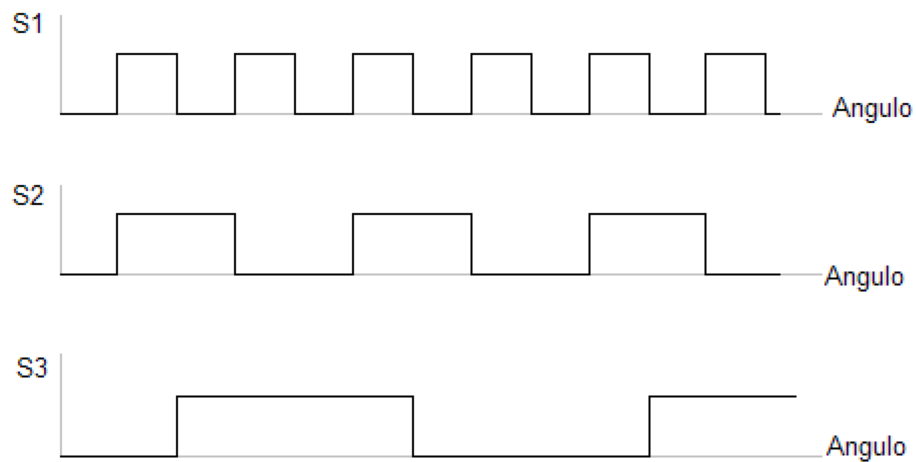


Figura 2-29 Encoder absoluto - Código Gray

A especificação de um encoder absoluto exige o número de bits de resolução, que corresponde ao número de trilhas perfuradas no disco sensor e das características mecânicas de fixação e proteção.

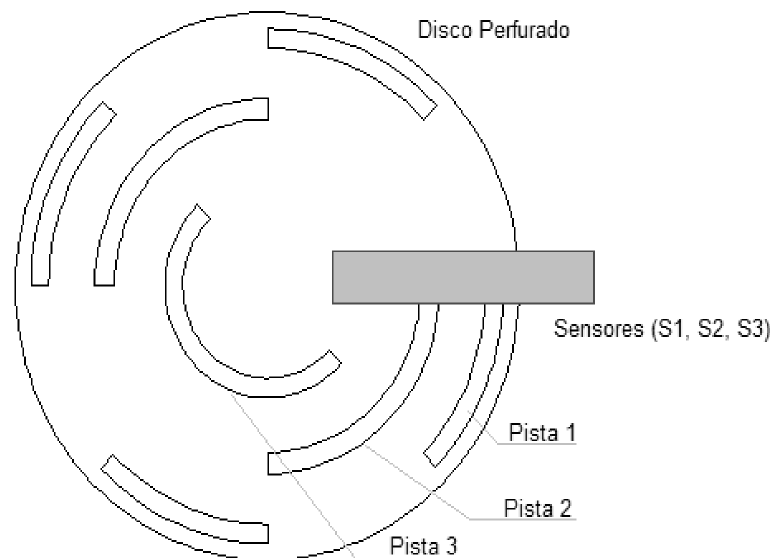


Figura 2-30 Encoder absoluto - Disco interno

2.4.5 Sensores Fotoelétricos

O sensor fotoelétrico é um sensor utilizado para identificar a presença de qualquer material. O funcionamento é baseado na utilização de um emissor e um receptor de luz, o objeto a ser controlado deve interferir na passagem e/ou

reflexão deste sinal luminoso. O receptor identifica o sinal luminoso e atua no circuito de saída do sensor.

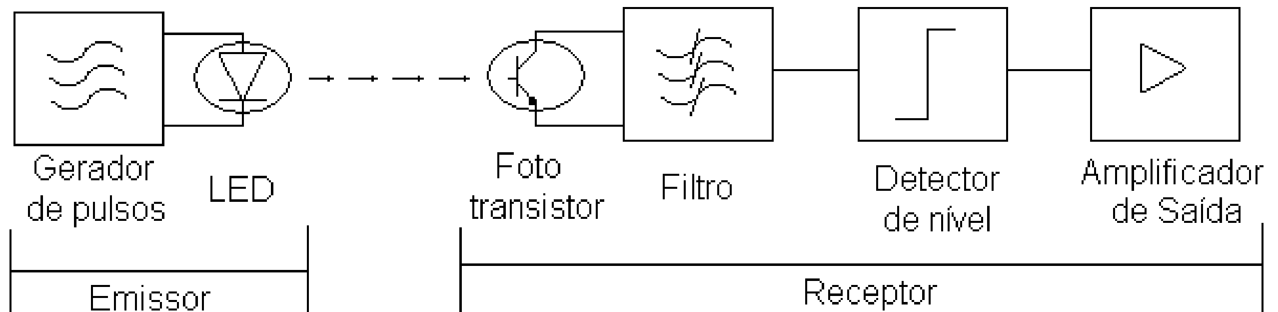


Figura 2-31 Diagrama de bloco - sensor fotoelétrico

Para evitar a interferência dos sinais luminosos externos, tais como a iluminação do pavilhão e luz natural solar, o circuito gerador de luz emite luz de forma pulsada evitando a interferência destes sinais externos.



Figura 2-32 Ilustrações sensores ótico

Estes sensores são construídos de três formas diferentes:

- Sensor ótico de barreira
- Sensor ótico difuso
- Sensor ótico refletivo

2.4.5.1 Sensores Óticos Barreira

Neste tipo de sensor o emissor e o receptor estão em unidades distintas e são dispostos um de frente ao outro garantindo a transmissão do feixe luminoso entre eles. O acionamento ocorrer quando o objeto interromper o feixe luminoso.

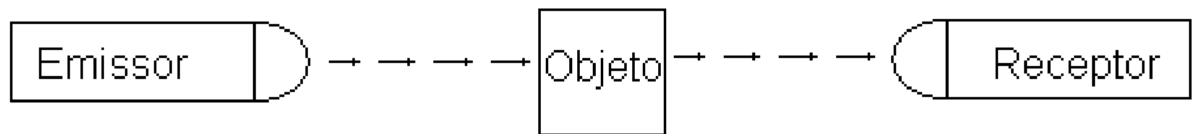


Figura 2-33 Representação sensor ótico barreira

2.4.5.2 Sensores Óticos Difuso

Neste tipo de sensor o emissor e o receptor estão em uma mesma unidade. O acionamento ocorrer quando o objeto estiver na região sensora do sensor refletindo o sinal através da superfície do objeto.

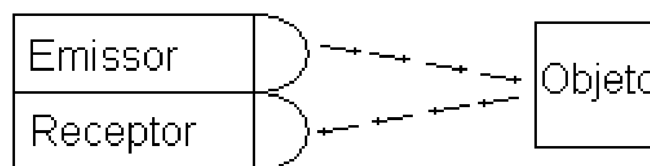


Figura 2-34 Representação sensor ótico difuso

2.4.5.3 Sensores Óticos Refletivo

Neste tipo de sensor o emissor e o receptor estão em uma mesma unidade, o feixe de luz enviado pelo emissor reflete em uma superfície prismática e retorna para o receptor. O acionamento ocorrer quando o objeto interromper este feixe luminoso.

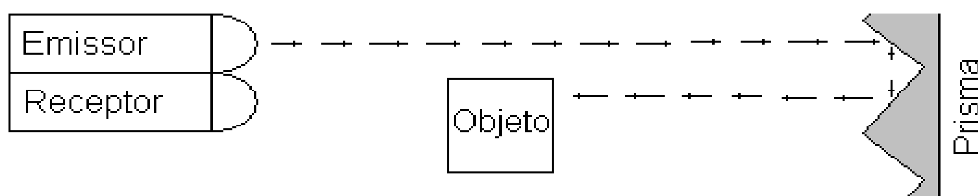


Figura 2-35 Representação sensor ótico refletivo

Para evitar à interferência da reflexão em superfícies refletoras que não sejam o prisma refletor, na maioria das marcar comerciais, a luz do sensor é polarizada, neste caso o sensor somente identificará a luz que for refletida no prisma, pois ao refletir-se no prisma ela envia luz em todas as direções, enquanto em uma superfície plana a luz é refletida com a mesma polaridade. O receptor está ajustado para receber o sinal da luz polarizada com 90° de defasagem do sinal emitido.

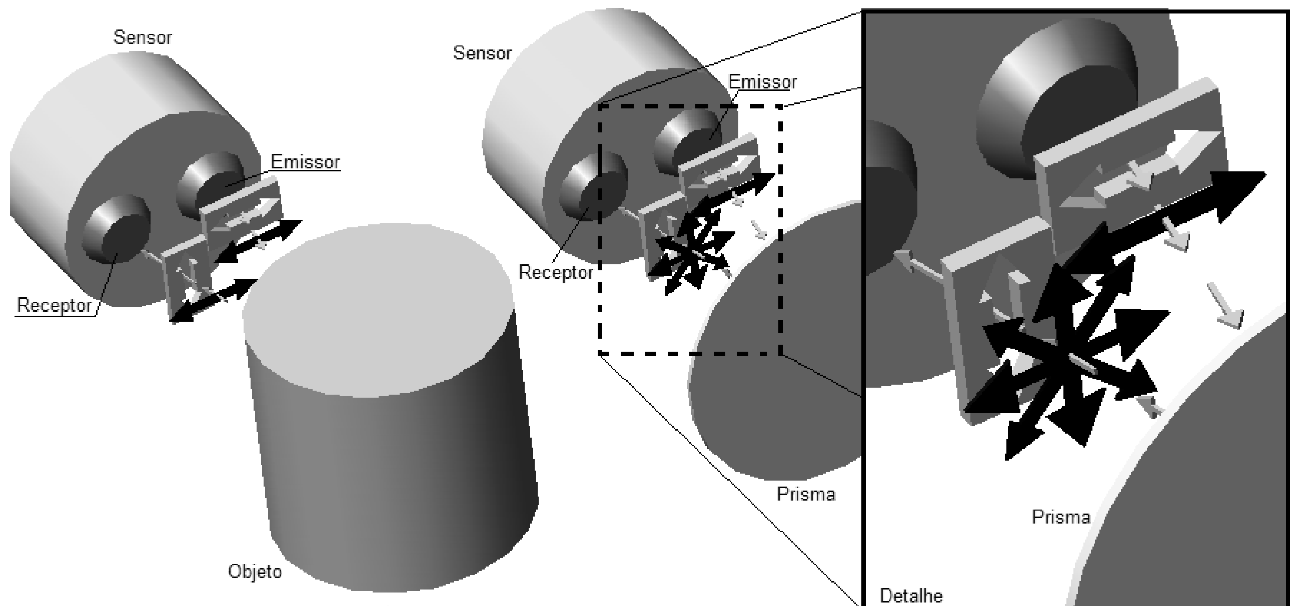


Figura 2-36 Polarização do sinal de Luz

2.4.6 Sensores de Fibra Ótica

Os sensores de barreira e difuso também são apresentados com a tecnologia da fibra ótica, podendo ser utilizado em local de difícil acesso e com alta sensibilidade de detecção de pequenos detalhes.

Uma aplicação do sensor de fibra ótica é a contagem de copos plásticos através da borda dos mesmos conforme mostra o diagrama da figura 2-31.

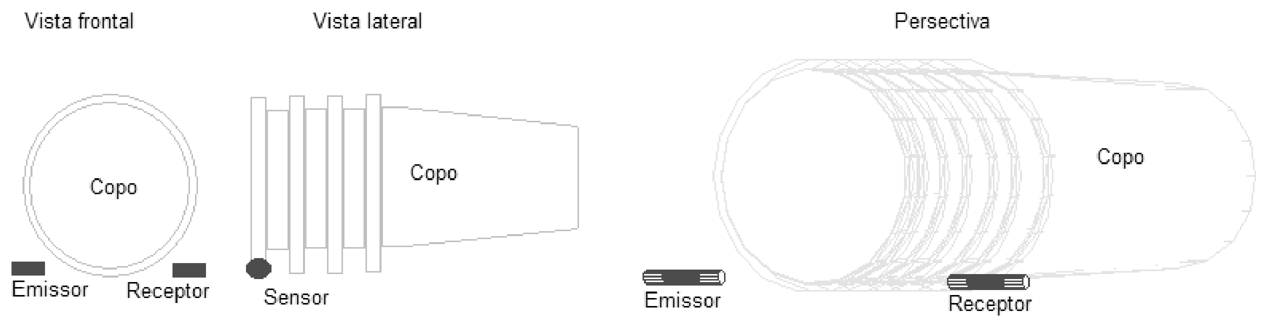


Figura 2-37 Exemplo utilização sensor fibra ótica de barreira

O circuito do sensor é composto de um driver para conversão dos sinais elétricos em óticos e vice-versa conforme diagrama da figura 2-32.

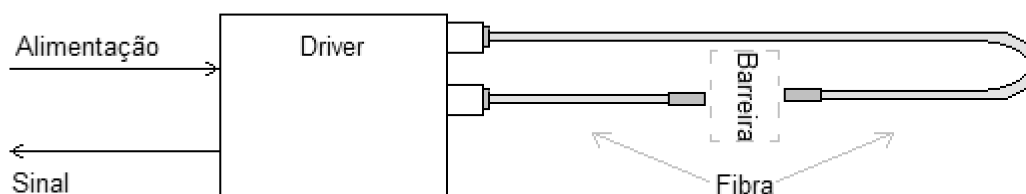


Figura 2-38 Diagrama elétrico sensor de fibra ótica

2.4.7 Sensores Indutivos

O sensor indutivo é utilização para identificar a aproximação de materiais ferromagnéticos. O funcionamento do sensor baseia-se na interação de um campo eletromagnético de alta freqüência com o material a ser identificado. O campo de alta freqüência é produzido por oscilador transistorizado e este campo é direcionado para fora dos elementos ativos do sensor formando uma região sensível à interferência magnética. A aproximação de um metal a esta região provoca a alteração do campo eletromagnético original ocasionando internamente a comutação da saída do sensor.

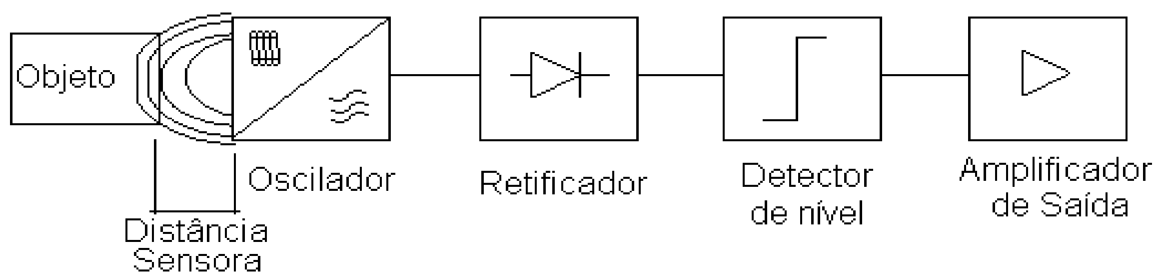


Figura 2-39 Diagrama de blocos - sensor Indutivo

A máxima distância de interação do campo magnético entre sensor e objeto é chamada de distância sensora.



Figura 2-40 Ilustração sensor indutivo

2.4.8 Sensores Magnéticos

O sensor magnético é utilização para identificar o posicionamento de certos atuadores. O funcionamento do sensor baseia-se na interação de um campo magnético fixo de um imã com uma micro-chave elétrica comutada por campo magnético. Este tipo de sensor é bastante utilizado para identificar o posicionamento dos atuadores pneumático, onde a chave é fixada na parede externa do cilindro e o imã está preso no embolo do pistão.

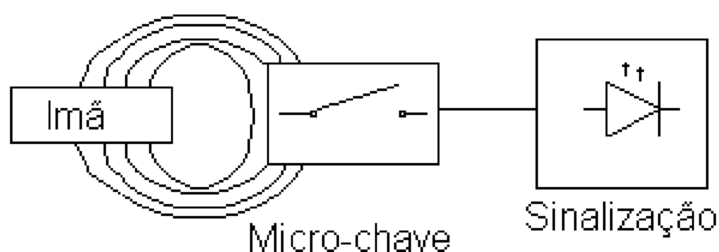


Figura 2-41 Diagrama de blocos - sensor magnético

Para facilitar a manutenção e identificar o funcionamento do sistema estes sensores também são providos de circuito de sinalização visual por led, para identificar o acionamento da chave.

2.4.9 Sensores Capacitivos

O sensor capacitivo de aproximação de materiais diversos é utilizado para identificar a presença de: materiais orgânicos, plásticos, pós, madeiras, líquidos, vidros, papéis, etc. O funcionamento deste sensor é baseado na interação do objeto com o dielétrico de um capacitor alimentado por sinal de alta freqüência. O objeto interagindo com o dielétrico do capacitor, varia a capacitância do elemento sensor, que por sua vez altera as condições do oscilador e aciona a saída do sensor.

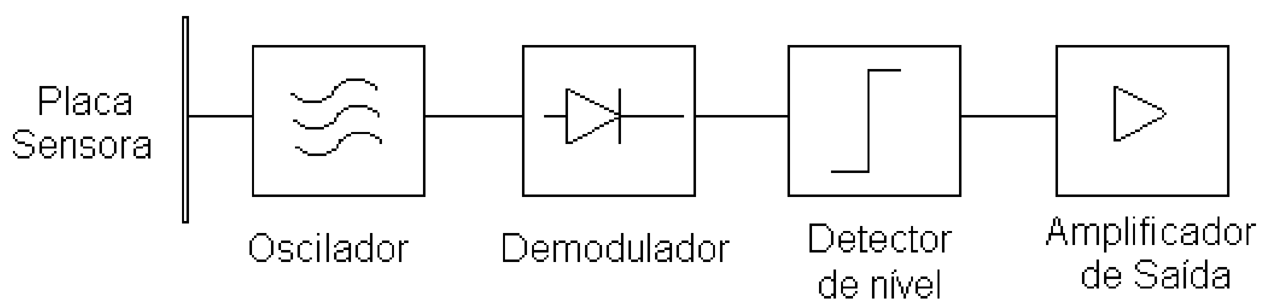


Figura 2-42 Diagrama de blocos - sensor capacitivo

A distância sensora depende do material e da massa a ser detectada, desta forma deve ser considerado um fator de redução para os diversos tipos de materiais. Este fator de redução e a distância básica são fornecidos nos catálogos dos fabricantes.



Figura 2-43 Ilustração sensor capacitivo

3 - Controlador Lógico

"No futebol, a competência do centroavante que imobiliza um contra-ataque está em desmarcar-se e também em pedir para que lhe passem a bola, em antecipar os movimentos da defesa, em ter cuidado com o impedimento, em ver a posição dos parceiros, em observar a atitude do goleiro adversário, em avaliar a distância até o gol, em imaginar uma estratégia para passar pela defesa, em localizar o árbitro, etc. Outros tantos esquemas podem ser trabalhados separadamente, no treino, mas um ataque eficaz dependerá da sua orquestração."

PERRENOUD, 1999, p.24

3.1 Definição

Um sistema automático é formando pelo conjunto entre controlador lógico e mecanismos de acionamentos. A integração destes dispositivos com o meio ocorre através de sinais elétricos de entrada provenientes de sensores, botoeiras, etc., e de dispositivos de saída, ou seja, atuadores, tais como: solenóides, válvulas, motores, contatores, lâmpadas, etc. A figura 3-1 mostra um diagrama com vários elementos utilizados em um sistema automático.

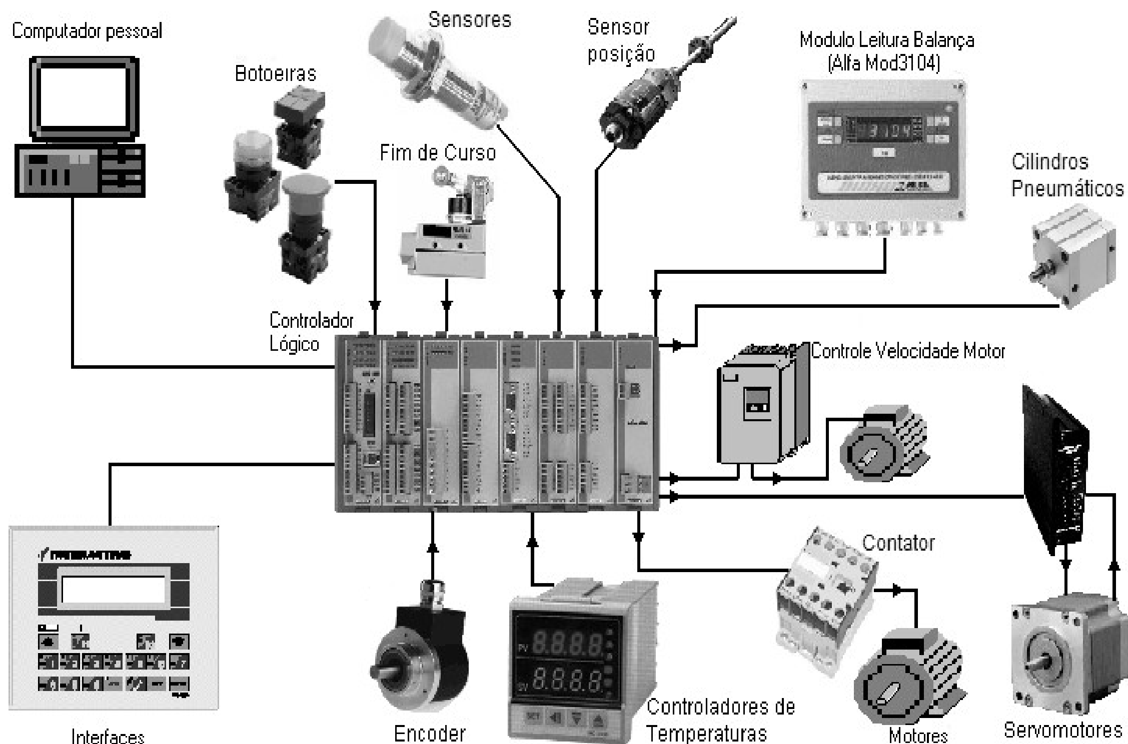


Figura 3-44 Representação controlador e periférico

No diagrama da figura 3-1 observamos que o controlador lógico é o elo de interligação entre os dispositivos de entrada e saída. Este controle é feito através de um sequenciamento de atividades pré-estabelecidas (programa) armazenado na memória do controlador. O funcionamento do sistema deve ser garantido pelo conjunto formado pela qualidade do controlador, atuadores elétricos e dispositivos de entrada utilizados no sistema.

Em 1978 a *National Electrical Manufactures Association* (NEMA), através do padrão NEMA ICS3-1978, determinou a seguinte definição para controlador lógico:

> "Equipamento de lógica digital, operando eletronicamente que usa memória programável para armazenamento interno das instruções de implementação específica, tais como: lógica seqüencial, temporização, contagem e operações aritméticas, para controle de máquinas e processos industriais com vários modelos de módulos de entradas e saídas digitais e analógicas em máquinas ou processos." (NEMA ICS3-1978)

3.2 Versatilidade do Controlador Lógico

O controlador lógico é muito utilizado em ambientes industriais, pois diminui a quantidade de fios e dispositivos utilizados nos painéis de controle, reduzindo o tempo e o custo de implantação de sistemas complexos, facilitando a manutenção e flexibilização de controle do sistema. São vantagens da utilização do controlador lógico em automação industrial:

- Velocidade de processamento (tempo de varredura do programa muito rápido);
- Tamanho reduzido;
- Baixo custo de aquisição;
- Flexibilidade para mudanças no programa de controle;
- Funções avançadas de programação;
- Possibilidade de comunicação com outros equipamentos;
- Diagnósticos de erros e monitoração do sistema de forma remota.

Desde o seu surgimento até hoje em dia, muita coisa evoluiu nos controladores lógicos como por exemplo: a variedade de tipos de entradas e saídas, o aumento da velocidade de processamento, a complexidade da linguagem de programação e a interface com o usuário.

3.3 Controlador Lógico

O controlador lógico é o coração do sistema, pois é ele quem vai gerir o funcionamento de todo o sistema. Neste trabalho utilizamos a denominação ***Controlador Lógico (CP)***. Em outras literaturas utilizam-se outras denominações, sendo que algumas delas são de propriedade intelectual reservada (patentes de fabricantes). As três principais denominações são:

➲ CP – controlador programável;
➲ CLP – Controlador Lógico Programável;
➲ PLC – Programmable Logic Controller.

O Controlador Lógico é um equipamento eletrônico formado por seis unidades principais, ou seja, processamento (CPU), de armazenamento de dados (memórias), entradas, saídas, comunicação serial e fonte de alimentação, como ilustra o diagrama da figura 3-2. Este diagrama de blocos de dispositivos é o mesmo utilizado para todos os circuitos de processamento de informações, tal como o micro-computador pessoal.

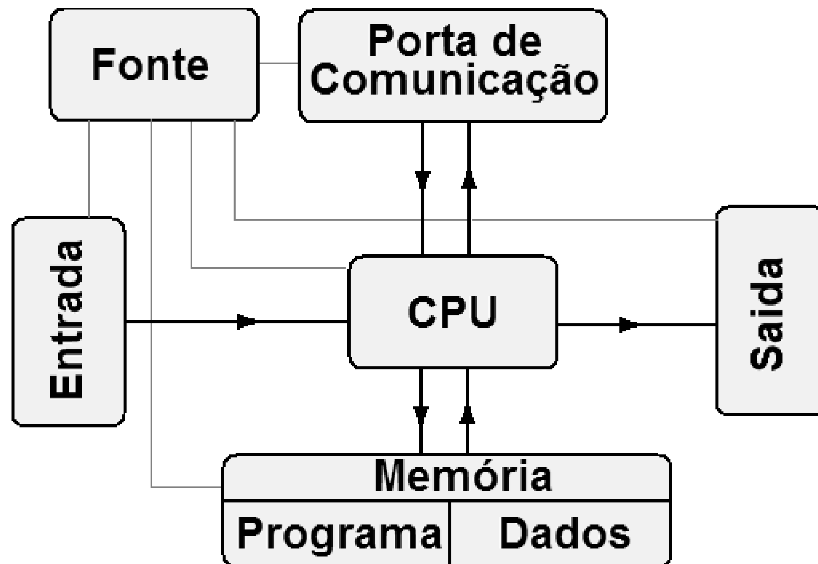


Figura 3-45 Diagrama em blocos

3.3.1 Unidade de Processamento (CPU)

A unidade de processamento é responsável pelo gerenciamento das atividades de um controlador lógico, realizando as atividades de interpretação do programa, leitura e escrita em memória e módulos de saídas. A unidade de processamento também é conhecida pela sigla de CPU ou UCP.

A unidade de processamento é composta por um componente eletrônico chamado microprocessador ou microcontrolador que definem as principais características desta unidade, tais como velocidade de operação, quantidade de dados para manipulação e funções disponíveis para programação.

3.3.2 Memórias

O sistema de memória de um controlador é responsável pelo armazenamento de dados e do programa. Existindo dois tipos de memórias, do ponto de vista construtivo: memórias RAM e memórias ROM.

RAM (*Random Access Memory*) são memórias de acesso aleatórios, para operação de leitura e escrita. Utilizada para armazenar dados temporários. A principal vantagem da memória RAM é a sua velocidade de acesso ser superior em comparação com memórias do tipo ROM. A principal desvantagem a volatilidade de seus dados, ou seja, os dados são perdidos com a desenergização da memória. Outra tecnologia de memória RAM é a NVRAM (*Non Volátil RAM*), ou seja, memória RAM com proteção dos dados, mesmo com o dispositivo desligado.

ROM (*Read Only Memory*) memórias apenas de leitura que são utilizadas para armazenar programas e constantes. Existem diversas tecnologias de memórias ROM, as principais são a EPROM e a FLASHEPROM. A EPROM (*Erasable Programmable ROM*), muito utilizada em máquinas antigas, é uma memória que permite a rescrita através de um programador, as principais desvantagens são: a necessidade da utilização de um programador exclusivo e de um apagador com luz ultravioleta. A memória FLASHEPROM é uma tecnologia mais recente que permite apagar e escrever através de sinais elétricos, sendo bastante utilizadas para armazenamento de programas aplicativos e dados dos controladores. A flexibilidade na manipulação de dados da memória FLASHEPROM permite que sejam também utilizadas para computadores tipo PC, através do canal USB em equipamentos chamados pendrives, pois permite o apagamento e a reescrita de informações através de sinais elétricos de forma simplificada.

Diz-se que a memória FLASHEPROM é apenas de leitura, pois a liberação de uma posição de memória utilizada não é permitida de forma individual. A liberação de posições de memórias deve ser feita em blocos, dependendo da tecnologia construtiva do dispositivo. Normalmente para as memórias utilizadas nos controladores lógicos o tamanho do bloco corresponde ao tamanho da própria memória e desde modo para liberar posições deve ser apagado todo o conteúdo da memória.

Nos controladores lógicos são utilizados os dois tipos de memórias RAM e FLASHEPROM. As memórias RAM são utilizadas para armazenar os valores

das variáveis do sistema e a FLASHEPROM o programa, constantes e configurações.

Para impedir que os dados da memória RAM sejam perdidos após uma desenergização, o controlador lógico possui um sistema especial de alimentação destas memórias através de uma bateria de Lithium, que a mantém sempre energizada. Através do software de programação definimos a região de dados para retenção dos dados, ou seja, que mantêm os dados intactos após a desenergização do sistema. Para garantir a troca de uma bateria descarregada ou defeituosa, também é utilizado um capacitor em paralelo com a bateria que mantêm o circuito energizado durante o processo de troca de bateria ou mesmo de uma falha momentânea da mesma. O circuito eletrônico simplificado desta alimentação elétrica está na figura 3-3

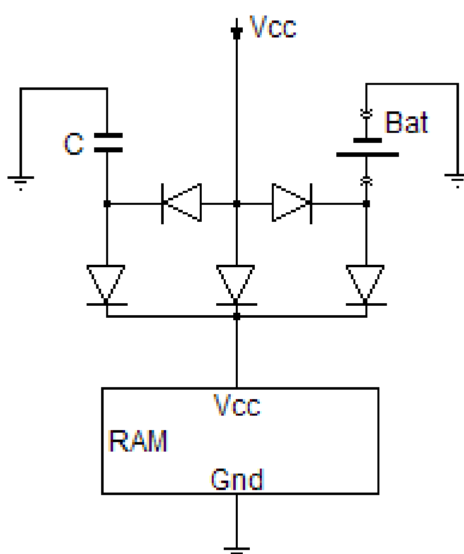


Figura 3-46 Alimentação da RAM

Para aumentar a velocidade de execução dos programas alguns controladores utilizam-se da memória RAM. Ao inicializar o sistema a CPU destes controladores primeiro cria uma cópia da FLASHEPROM para a RAM, executando o programa na memória RAM a partir deste momento. Na figura 3-4 observamos que o programa aplicativo fica armazenado nas duas memórias, mas apenas o programa da FLASHEPROM fica armazenado após a desenergização do sistema.

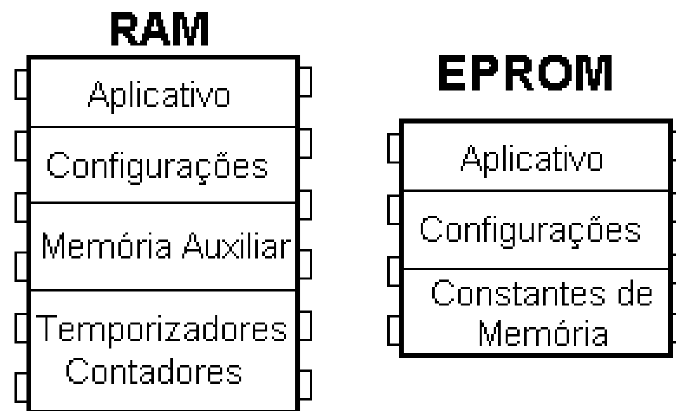


Figura 3-47 Mapa de memória

É oportuno observar que alguns fabricantes para diminuir o custo final do equipamento colocam como acessório a memória FLASHEPROM e o controlador lógico trabalha apenas com a memória RAM. Neste caso o programa será perdido caso ocorra ao mesmo tempo uma falha na bateria interna e na alimentação da rede elétrica.

3.3.3 Unidades de Entradas e Saídas

As unidades de entradas e saídas (E/S), são as unidades que fazem o interfaceamento do processador central com o meio externo (Processo/máquina). Sua construção eletrônica é robusta a fim de isolar os ruídos elétricos e garantir o funcionamento dos equipamentos conectados ao sistema.

As unidades de entradas recebem os sinais externos, provenientes do sistema controlado, isolando-os e ao mesmo tempo normalizando-os para a interpretação da unidade de processamento. As unidades de saídas são responsáveis por entregar o sinal da CPU para o meio externo, garantindo a isolação, normalização e amplificação deste sinal.

3.3.3.1Padrões de Sinais Digitais

Para representar o sinal digital nos circuitos físicos industriais, são utilizados alguns padrões de níveis de tensão, conforme citado abaixo:

- ➲ 24 V_{DC} – Padrão mais utilizado no ambiente industrial;
- ➲ 125 V_{DC};
- ➲ 110 V_{AC} – Padrão de sinal para alimentação direta da tensão de rede Alternada;
- ➲ 220 V_{AC}.

3.3.3.2 Padrão Construtivo Entrada

As entradas são classificadas quanto à tecnologia empregada para sua construção, como:

- Transistor NPN – tecnologia que utiliza o sinal positivo como referência de tensão, sendo o sinal negativo interpretado pelo controlador . Veja que na figura 3-5 o valor entregue para a carga é o sinal negativo (zero volt) enquanto o sinal que permanece como referência é o sinal positivo (24 V).

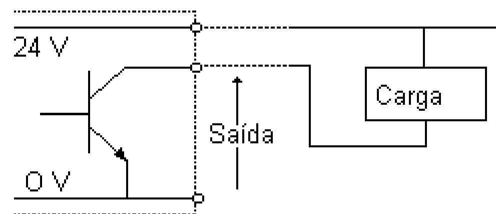


Figura 3-48 Circuito NPN

- Transistor PNP – tecnologia que utiliza o sinal negativo como referência de tensão. O sinal interpretado no controlador é o sinal positivo.

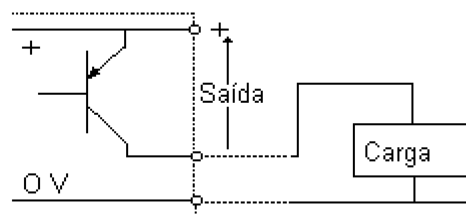


Figura 3-49 Circuito PNP

Além dos padrões construtivos, as entradas dos controladores são disponibilizadas com isolação entre o circuito externo e interno utilizando-se acopladores óticos. Devido a esta forma construtiva a maioria dos fabricantes fornece o controlador com possibilidade de configuração do tipo de entrada NPN-PNP conforme conexão do sinal de referência. Na figura 3-7 visualizamos a conexão deste sinal de referência de um controlador para configurá-lo em NPN e PNP.

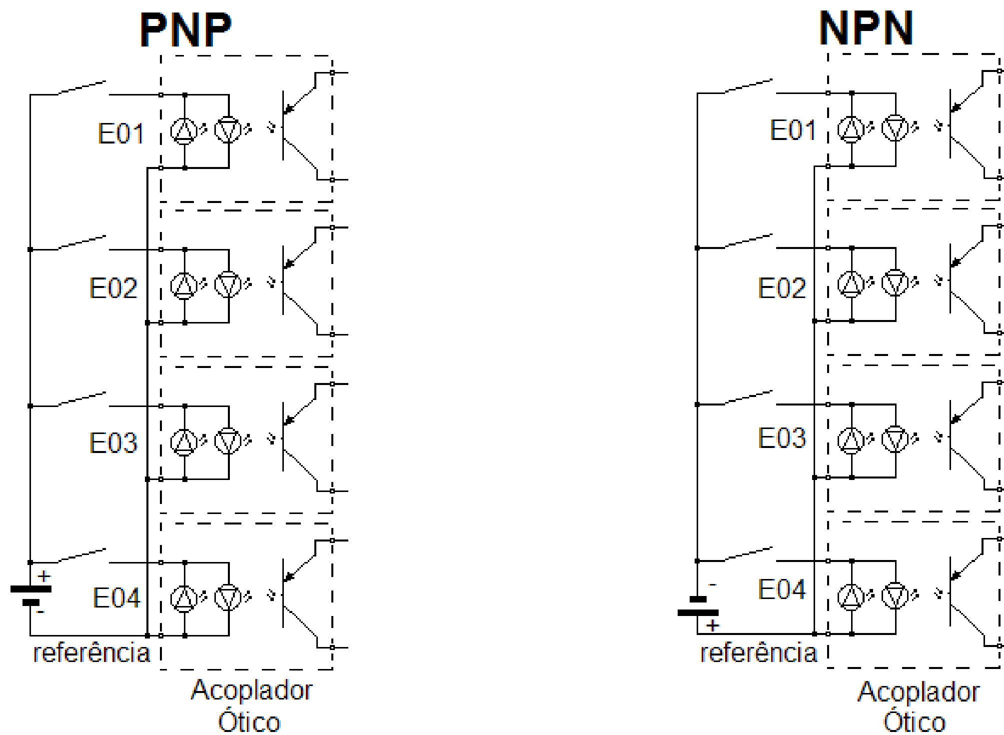


Figura 3-50 Entradas NPN e PNP

3.3.3.3Padrão Construtivo Saída

A principal função das saídas é entregar energia suficiente para acionar as cargas conectadas ao controlador. Portando a análise da forma construtiva das saídas é considerada sobre esta ótica. As saídas são construídas como:

- Transistor – A comutação do sinal é feito com um elemento semicondutor. A vida útil do sistema é maior, devido à ausência de partes móveis. Desvantagem: baixa capacidade de corrente. O circuito é representado nas figuras 3-5 e 3-6.

- Saída a Relé – tecnologia que utiliza um contato auxiliar de um relê como sinal de saída, apresentando uma corrente de saída maior. Desvantagem: seu contato é móvel, ou seja, vida útil menor. O circuito é representado na figura 3-8.

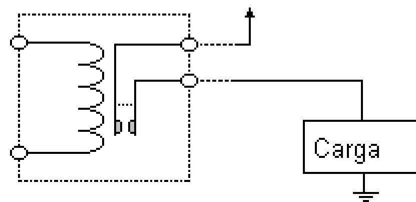


Figura 3-51 Saída a rele

- Tiristores – ou relê de estado sólido, tecnologia que utiliza tiristores ou transistores de potência para a saída do sinal. Vantagens: Corrente de saída maior, normalmente 5 Amperes e Vida útil maior pois não apresenta partes móveis. A figura 3-9 representa o circuito.

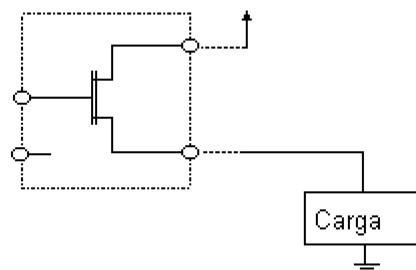


Figura 3-52 Rele Estado Sólido (SSR)

A tabela 3.1 faz uma comparação entre estas tecnologias:

Tecnologia	Vida útil	Corrente saída	Desvantagem
Transistor	Ótima	200 mA	Baixa corrente
Rele	Boa	2 A a 10 A	Vida útil
Tiristor	Ótima	5 A	Custo maior

			Pouca opção de mercado

Tabela 3-2 Comparação entre tecnologias

3.3.3.4Padrão Sinal Analógico

Os sinais elétricos analógicos são utilizados com alguns padrões, nível e grandeza elétrica para ser utilizado em ambiente industrial, como segue:

- ➲ 0 a 10 V – bastante utilizado devido à facilidade técnica de implementação, com uma fonte de alimentação e um potenciômetro como meio para variar a tensão;
- ➲ 0 a 5 V – Variante do método 0 a 10 V, mas pouco utilizado.
- ➲ 1 a 5 V – Variante do método de tensão, também pouco utilizado, permite identificar falhas na conexão, pois a tensão mínima é diferente de zero.
- ➲ -10 a 10 V – Método 0 a 10 V modificado para utilização em controle de velocidade, permitindo a reversão do sentido de giro de um motor.
- ➲ 4 a 20 mA – A padronização da corrente elétrica torna os sistemas de leituras de grandezas analógicas mais precisas, pois não sofrem interferência da resistência dos condutores. Este método é bastante utilizado pois permite identificar um circuito defeituoso, como um cabo rompido.
- ➲ 0 – 20 mA - circuito de corrente simples que não permite identificar falha no condutor. Método pouco utilizado.

Para utilizar os sinais analógicos são tomados alguns cuidados na instalação, evitando a interferência de distúrbios e falhas elétricas externas, utilizando para este fim cabos blindados, canaletas/tubulações exclusivas, um bom sistema de aterramento e afastar os condutores de sinais dos condutores dos circuitos de força, além de verificar todas as sugestões definidas pelo fabricante.

3.3.4 Fonte de Alimentação

A fonte de alimentação é um dispositivo eletrônico capaz de transformar e manter a energia elétrica dentro dos padrões requeridos pelo sistema. Em um sistema automático com controlador lógico a fonte de alimentação é responsável pelo fornecimento de energia em um nível de tensão adequado. Este nível de tensão, normalmente, é 24 Volts. A especificação da fonte deve estar de acordo com a potência elétrica requerida pelo sistema, ou seja, a corrente de saída máxima da fonte deve ser maior que a corrente máxima de funcionamento.

É importante observar que a corrente máxima de funcionamento do sistema nem sempre é igual à soma das correntes de todos os dispositivos do conjunto. Isto ocorre devido a não simultaneidade dos acionamentos, existindo um fator de diversidade entre eles. O fator de diversidade deve ser previsto pelo projetista levando em conta o momento em que o maior número de dispositivos está acionado ao mesmo tempo.

Uma fonte de alimentação é especificada pela tensão (E) e potência (P), e sua corrente elétrica (I). A relação entre elas é definida pela seguinte formula:

$$P = E \cdot I$$

As fontes de alimentação são construídas com base em diversas tecnologias para manter o nível de tensão contínuo. Estas tecnologias são:

- Convencional - é a mais simples de todas, apresenta como desvantagem uma oscilação em torno do valor de saída, normalmente +/- 10%.
- Regulada – O valor de saída é constante, mas apresentam, normalmente, baixa capacidade de corrente de saída.
- Chaveada – O valor de saída é constante, apresentando como vantagens o tamanho reduzido e a alta capacidade de corrente.

3.3.5 Porta de Comunicação

As portas de comunicações são os meios oferecidos para a troca de informações com o meio externo. Através de uma porta de comunicação

conectamos o controlador lógico com um terminal programador, com uma IHM, com uma rede de controladores e dispositivos ou até mesmo com um programa de supervisão.

O tipo de comunicação utilizada é normalmente a serial e existem diversos tipos de protocolos de comunicação, bem como tecnologias de fabricação das portas que serão tratadas no capítulo 6 – Comunicação com CP.

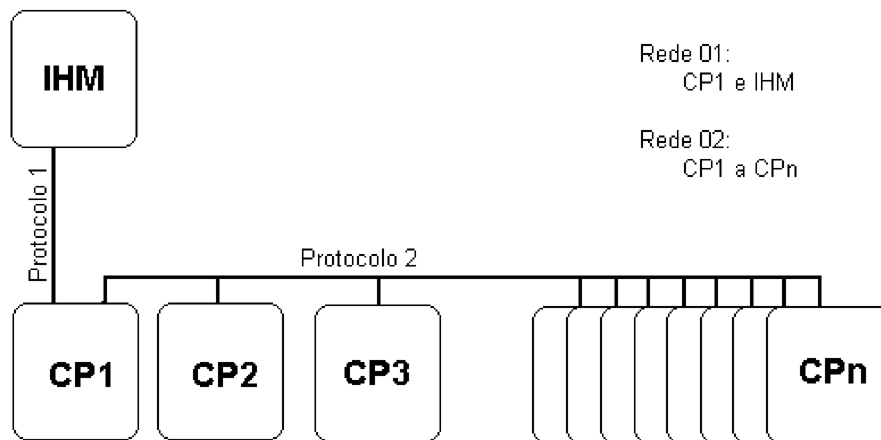


Figura 3-53 Utilização de portas de comunicação

Na especificação do controlador lógico consideramos também a necessidade de portas de comunicações extras, normalmente um controlador vem com uma ou duas portas de comunicação. Lembrando-se que uma mesma porta de comunicação não é utilizada para duas funções diferentes ao mesmo tempo. A figura 3-10 apresenta um diagrama que mostra um controlador conectado a duas redes de comunicação diferente, neste caso o CP1 possui duas portas de comunicação.

3.4 Projeto Elétrico

Para a realização da automação com qualidade e segurança, e minimizando o tempo de manutenção no futuro, o sistema de automação deve se acompanhado de um projeto elétrico do sistema. Neste projeto elétrico devem constar todos os elementos utilizados para o acionamento e proteção do sistema elétrico, além de obedecer a todas as normas de seguranças vigentes.

O diagrama elétrico é um desenho que representa através de símbolos todos os elementos que compõem o sistema elétrico. O diagrama deve ser acompanhado de sua simbologia para que o diagrama seja compreendido.

3.4.1 Exemplo Projeto Elétrico

Desenhar o diagrama elétrico de um sistema de controle de nível representado na figura 3.11. Onde temos:

- LH – *Level High* – sensor de contato para nível alto;
- LL - *Level Low* – sensor de contato para nível baixo;
- B1 e B2 – MotoBombas com motor de 380 V / 60 Hz e comando em 220 V;
- V1 e V2 – Válvulas eletro pneumáticas com comando de 24 V.

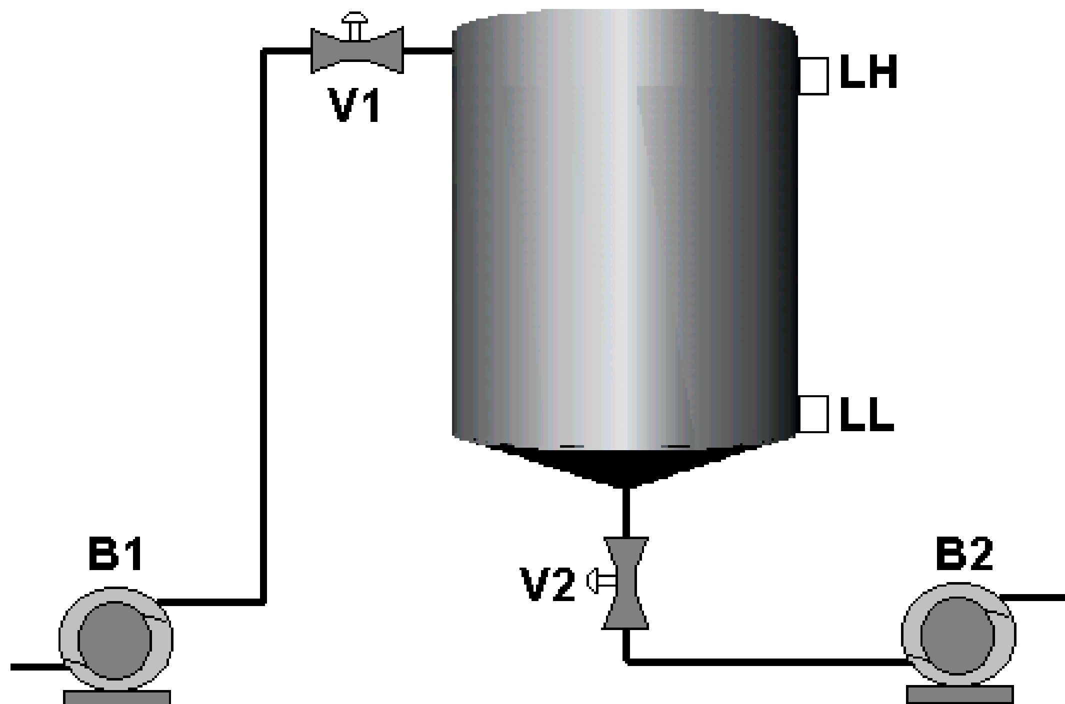


Figura 3-54 Controle de nível

Para o sistema proposto o projeto foi subdividido em duas partes, o circuito de comando e o circuito de força. O circuito de comando tem o acionamento dos equipamentos e dispositivos de comando e controle, como

ilustra a figura 3-12. Na figura 3-13 temos o circuito de força onde aparece a proteção geral do circuito e alimentação efetiva dos motores através dos contatores que foram acionados no circuito anterior. Em ambos os circuitos observamos uma linha pontilhada indicando “campo”, pois neste espaço estão colocados os dispositivos que não fazem parte do painel elétrico e estão instalados em outros setores da fábrica ou máquina. Em nosso circuito elétrico devemos especificar no projeto a localização dos dispositivos que serão instalados fora do painel, tais como: campo, entrada, saída, expedição, etc. A simbologia utilizada para este circuito aparece na figura 3-14.

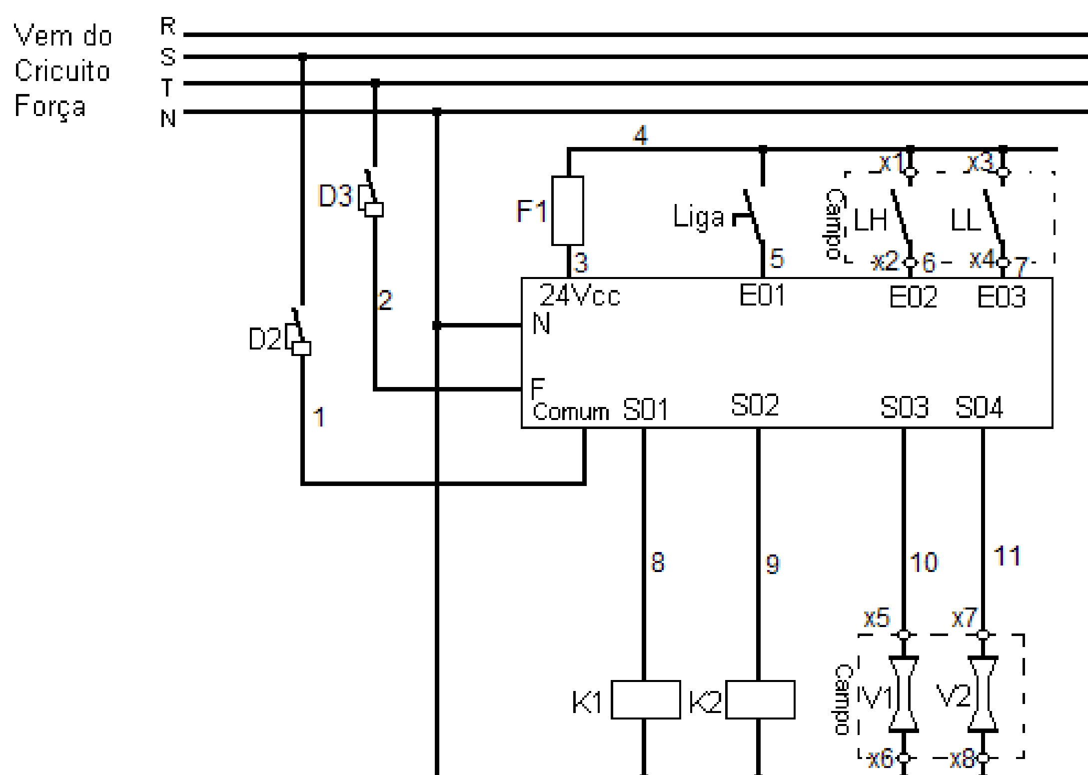


Figura 3-55 Circuito de comando: controle de nível

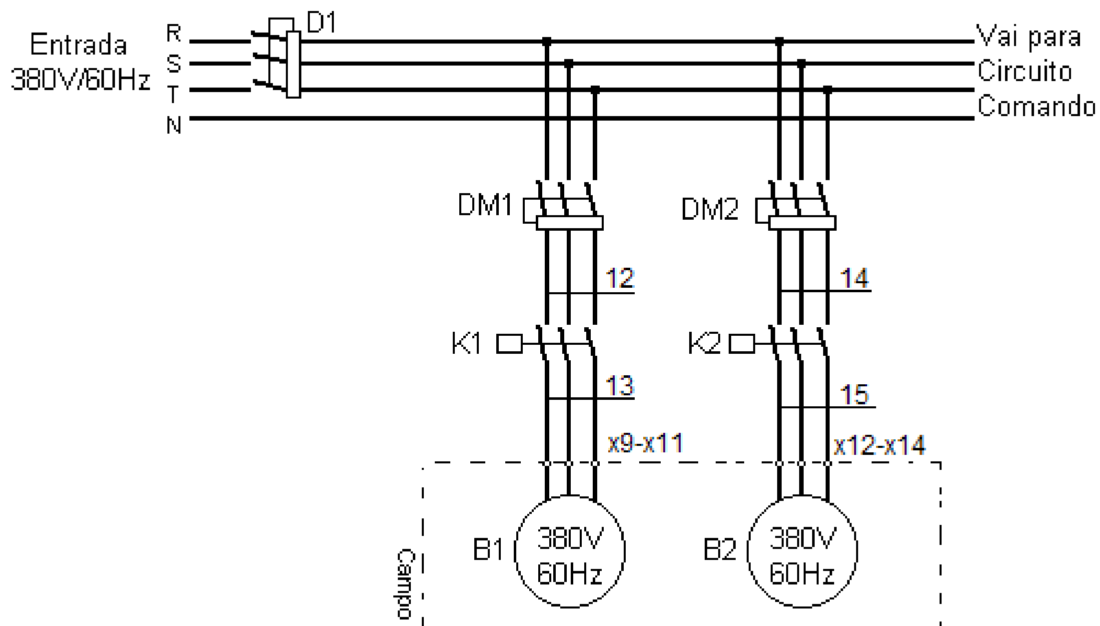


Figura 3-56 Circuito de força: controle de nível

Para facilitar a conexão elétrica entre os dispositivos instalados internamente no painel elétrico e o ambiente industrial, são utilizados pontos de conexão elétrica, chamados bornes de conexão (figura 3-15), estes bornes estão representados no diagrama (figuras 3-12 e 3-13) através do pequeno circulo entre a linha pontilhada. Estes bornes são numerados conforme a seqüência de instalação no painel elétrico.

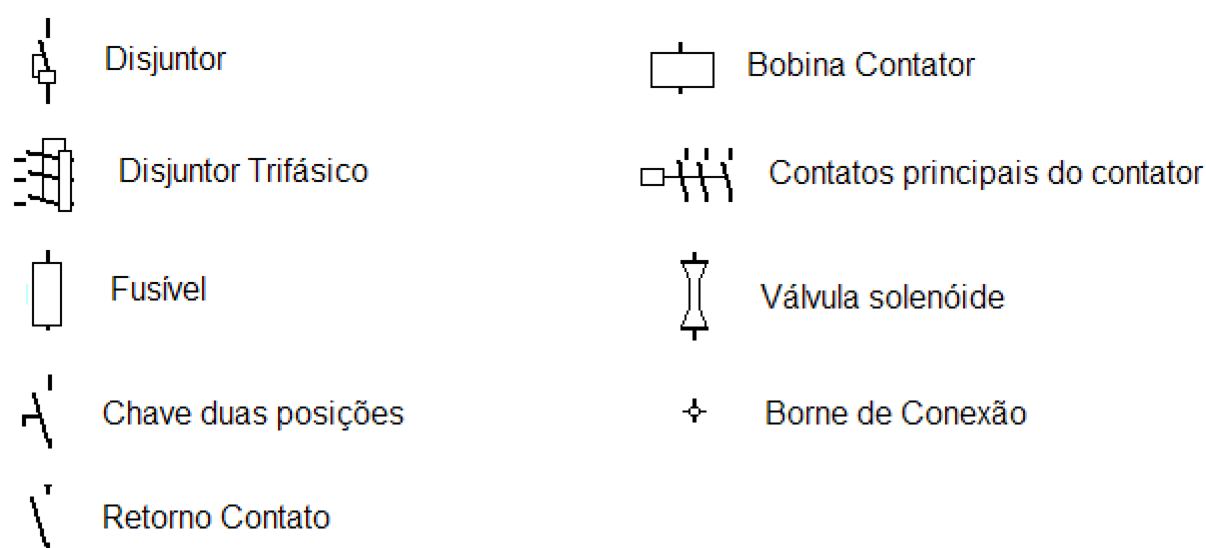


Figura 3-57 Simbologia diagrama elétrico

Para facilitar o trabalho de manutenção todos os fios utilizados internamente no painel estes são numerados com uma identificação exclusiva (figura 3-16), esta numeração é conhecida por anilha. Nos diagramas das figuras 3-12 e 3-13 a numeração de cada fio aparece ao lado da linha que representa o condutor.

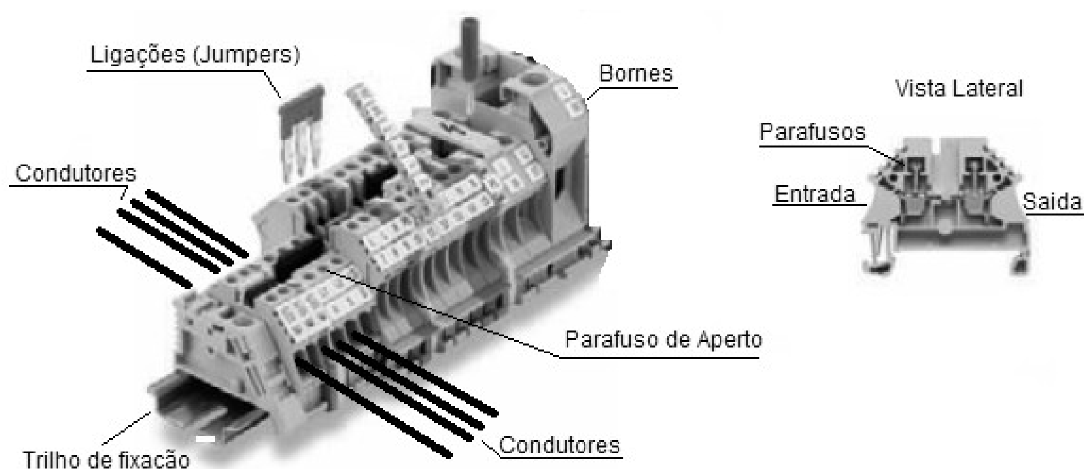


Figura 3-58 Bornes de conexão

Figura 3-59 Sistema de numeração (anilha) de condutores

3.4.2 Exercício Proposto: Controle de Linha Cerâmica

Desenhar o diagrama elétrico de um sistema de controle de uma linha cerâmica representado na figura 3-17. Os equipamentos instalados com seus respectivos sinais de controle são relacionados abaixo:

- T1 – Tração um. Este sistema é formado por um motor controlado por um inversor de freqüência, e neste estágio temos: sinal de controle para habilitar o inversor de freqüência. Uma chave de emergência para bloqueio de todo o sistema. Retorno do inversor de freqüência via saída a rele do inversor.
- T2, T3 e T4 – Tração 2, 3 e 4 semelhante à tração 1 com os mesmos sinais.
- S01 – Serigráfica 1- A serigráfica constitui-se em um circuito de controle independente e necessita para seu funcionamento um sinal de controle, indicando que a linha esta em funcionamento, ao mesmo tempo em que libera para a linha um sinal de retorno indicando que está desbloqueada, ou seja, pronta para o trabalho.
- S02 e S03 – Serigráfica 02 e 03, semelhante à serigráfica 01.
- H01 – Habilita sistema, sinal de consenso proveniente do inicio da linha que indica que o sistema deve trabalhar.

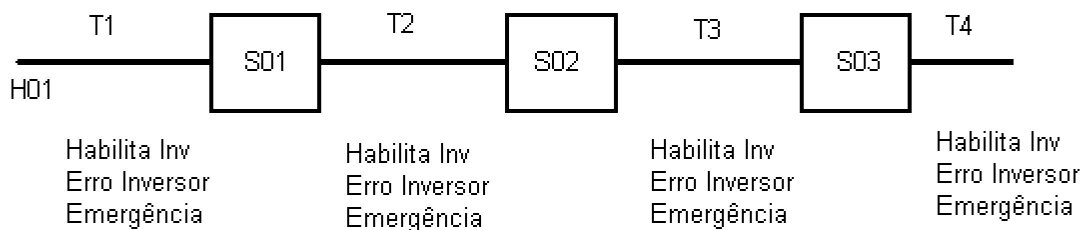


Figura 3-60 Linha industrial cerâmica

A figura 3-18 representa tridimensionalmente a linha cerâmica representada no layout da figura 3-17.

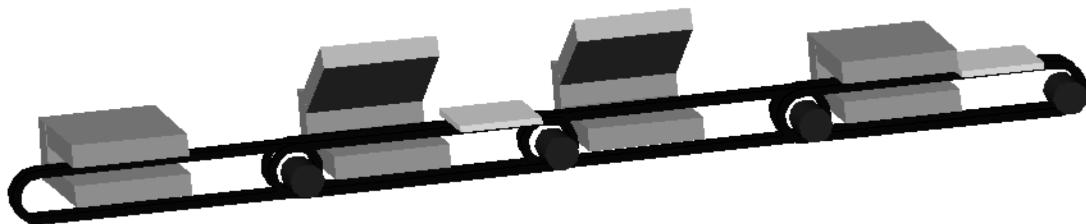


Figura 3-61 Linha industrial cerâmica: vista 3D

3.5 Configuração

Configuração de um controlador lógico é o processo de alocação de periféricos junto ao barramento da CPU com a utilização do software de programação. Devemos configurar os dispositivos de entradas e saídas, digitais e analógicos e a utilização da porta de comunicação. Na configuração do sistema de controle são configurados:

➲ Módulos de entradas e saídas não integrados a CPU;

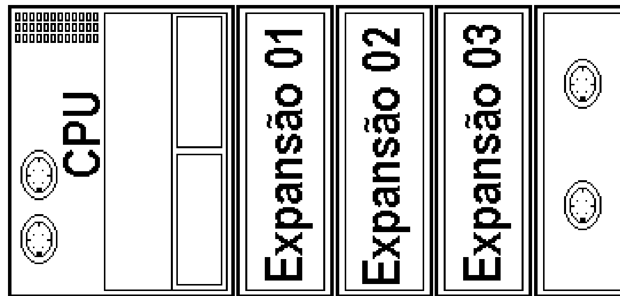


Figura 3-62 Representação de controlador com expansão

- Módulos de entradas e saídas analógicos, definindo o range de leitura e a possibilidade de utilização de filtros de leituras;
- Módulos de contagem rápida, definindo o tipo de leitura e precisão;
- Quantidade de memória retentiva;
- Retentividade dos pontos de entradas e saídas;
- Parâmetros das portas de comunicação (velocidade, bit de controle, etc.);
- Tempo de execução da interrupção de tempo.

Alguns controladores não necessitam de configuração adicional, pois os endereços utilizados pelos periféricos já vêm pré-definidos.

3.6 Programa Gerenciador

Para gerenciar os dispositivos internos de um controlador com a execução do programa do usuário, existe um pequeno programa, interno e transparente ao usuário, desenvolvido pelos fabricantes. Este programa é responsável pela integração de todos os módulos de um controlador (memória, E/S, porta de comunicação), verificação de erros, pela leitura das unidades de entrada, pela escrita nas unidades de saídas e pelo controle das memórias. Este programa é conhecido por *firmware* e em língua portuguesa *de programa executivo* ou *programa gerenciador*. Devemos notar que em nosso microcomputador de mesa o executivo é a *Bios* (*Basic Input / Output System*) do sistema.

O fluxograma do ciclo de varredura é visualizado na figura 3-20.

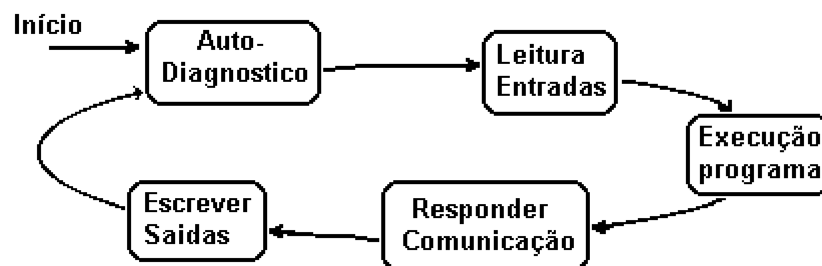


Figura 3-63 Ciclo de varredura (Scan)

Neste fluxograma o programa executivo realiza as ações descritas na tabela 3-2 durante um período completo (SCAN) de leitura de um controlador.

Passo	Descrição	Observação
01	Energização	
02	Processo de Energização	Verifica memória e programa do usuário
03	Verifica modo de operação	Se modo RUN continua se STOP vá para passo 07
04	Inicialização de dados	Verifica Hardware e inicializa memórias
05	Atualiza Entrada	Atualiza mapa espelho dos pontos de entradas
06	Verifica programa	Executa uma varredura no programa
07	Verifica comunicação	Verifica dado via serial
08	Atualiza Saídas	Atualiza saídas com referência ao mapa espelho
09	Inicializa Watch-Dog	Inicializa pulso do alarme de watch-dog
10	Reinicializa	Se modo RUN vá ao passo 5 se modo STOP vá ao 3

Tabela 3-3 Ciclo de scan

3.7 Tipos de Dados para Processamento

Em um sistema de processamento de dados, tal qual se propõe o controlador lógico, o acesso e manipulação dos dados são feitos com base nos tipos de dados disponibilizados para acesso. O sistema com controlador oferece diversos tipos de dados para ser utilizado nos programas que serão

desenvolvidos. Na tabela 3-3 relacionamos o tipo de dado, sua descrição e range de informações que ele armazenar.

Tipo	Descrição	Range
Bit	Menor unidade de informação de acesso binário	0 ou 1
Nibble	Conjunto de quatro bits	0 a 15
BYTE	Conjunto de oito bits. Esta é a unidade de divisão da capacidade dos circuitos de memórias.	0 a 255
WORD SIGNED	Conjunto de 16 bits, com representação do sinal.	-32768 a 32767
WORD UNSIGNED	Conjunto de 16 bits, sem a representação do sinal.	0 a 65535
DOUBLE WORD	Conjunto de 32 bits. Seu valor representar um número na base decimal ou na base BCD	-2.147.483.648 a +2.147.483.647.
PONTO FLUTUANTE	É um tipo de operando que permitem representar número fracionário. A utilização deste tipo de operando aumenta a precisão dos cálculos aritméticos, já que os operandos anteriores trabalham apenas com números inteiros. Estes números são representados da seguinte forma: $0{,}MATRIZ \times 10^{EXPOENTE}$ A matriz e o expoente são armazenados em operandos de 16 ou 32 bits, dependendo do fabricante do controlador lógico.	

Tabela 3-4 Tipo de dados para processamento

3.8 Mapeamento de Memória e Dispositivos

A manipulação dos dados de memórias e acesso aos dispositivos de entrada e saída ocorre através de um endereçamento lógico que denominamos de mapeamento de memórias e dispositivos. Podemos subdividir o mapeamento em dispositivo de entradas digitais, dispositivos de saída digitais, reles auxiliares e registradores, estes dois últimos sendo exclusivamente posições de memórias.

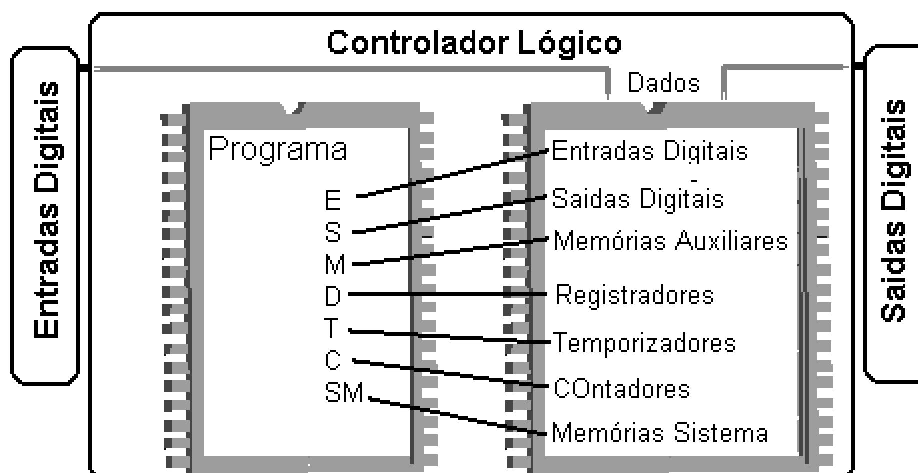


Figura 3-64 Endereçamento de memória

O acesso a uma posição especifica da memória ocorre através de um endereço que referencia a posição e a região de memória. Os dispositivos de entradas digitais, bem como os de saídas, são representados normalmente por uma letra, que representa entrada ou saída, mais um número sequencial representando o endereço do byte ou word do bloco de memória apontado. Como segue nos exemplos abaixo:

Endereço	Descrição
X1 a X125	Entrada digital CP TP02 WEG, contagem continua a partir de X1.
Y1 a Y125	Saída digital CP TP02 WEG, contagem continua a partir de Y1.
E0.0 a E50.7	Entrada digital CP Altus, grupos divididos em octetos, a partir de E0.
S10.0 a S50.7	Saída digital CP Altus, grupos divididos em octetos, a partir do último número reservado para a entrada. Via software, o programador, pode definir o número inicial para a saída.
0.00 a 19.15	Entrada digital no OMRON, numerado de forma continua a Cada bloco de 16 bits.
10.00 a 10.00	Saída digital no OMRON, endereçamento continuo a cada grupo de 16 bits, para a série CPM2A
I0.0 a I7.7	Entradas digitais para CP Siemens, endereçamento por octeto.
Q0.0 a Q7.7	Saídas digitais para CP Siemens, endereçamento por octeto.
I1.0 a I3.16	Entradas digitais CP Telemecanique, endereçamento de cada bloco de

	E/S a partir do endereço zero.

Tabela 3-5 Endereços de entradas digitais

As memórias auxiliares são endereçadas de diversas formas: através de bit ou de grupos de 16 bits. Cada fabricante define o seu próprio sistema de endereçamento, segue tabela 3-5 alguns exemplos.

Fabricante	Endereço	Descrição
WEG	C1 a C1024	Relê auxiliar
	V1 a V1024	Registrador de 16 bits
ALTUS	M0.0 a M1024.f A0.0 a A95.7	Reles auxiliares em grupo de 16 bit Ou grupos de 8 bits
	M0 a M1024 D0 a D1024	Registradores de 16 bits Registradores de 32 bits (BCD)
OMRON	IR0.00 a IR1000.15	Reles auxiliares
	DM0 a DM6000	Registradores de 16 bits Aos pares registradores de 32 bits
Telemecanique	%M0 a %M200	Reles Auxiliares
	%Mw0 a %Mw500	Registradores de 16 bits
Siemens	M0.0 a M50.7	Reles Auxiliares
	Mw0 a Mw50	Registradores de 16 bits

Tabela 3-6 Endereço das saídas digitais

Alguns controladores reservam um espaço da memória, para armazenar dados referentes ao processamento interno do sistema. Estes dados podem ser utilizados no programa em ladder. A esta região de memória chamamos de memória de sistema. Nesta região de memória são armazenados dados tais como: tempo de scan, sinalizadores de comparação, pulsos auxiliares de tempo, etc. Para entender o que significa cada bit e cada registrador especial deve-se consultar o manual do fabricante. Na tabela 3-6 mostramos alguns exemplos.

Endereço	Descrição
M8002	Flag de energização no CP Mitshubishi
SC58	Flag de energização no CP WEG
	O Altus não possui flag de energização, ele utiliza um procedimento de energização: E-XXXXXX.000
SR253.15	Flag de energização do CP OMRON
S0.3	Flag de energização do CP Siemens
%S13	Flag de energização do CP Telemecanique

Tabela 3-7 Endereços de memórias auxiliares do sistema

O endereçamento de dispositivos de entrada analógica depende de cada fabricante, e cada um deles tem uma forma especial de transformar os dados presentes na entrada em um sinal digital de 16 bits, ou mesmo de forma contrária, enviando os dados para uma saída analógica. O endereçamento e manipulação de dados de uma interface dependem também de tecnologias adotadas pelos fabricantes. Recomendamos nestes casos verificar os respectivos manuais dos fabricantes.

A memória RAM será o dispositivo de origem destes dados e o apontador normalmente estará fazendo referência direta a uma posição desta memória. Salientamos que para aumentar a velocidade de processamento a CPU do controlador realiza uma cópia de trabalho do programa do usuário na memória RAM. O mesmo ocorre no gerenciamento dos dispositivos de entradas e saídas, os quais o controlador faz acesso apenas uma vez no ciclo de funcionamento (scan), criando uma imagem destes dados para trabalho. A figura 3-22 apresenta um diagrama que mostra a organização interna de uma memória RAM.

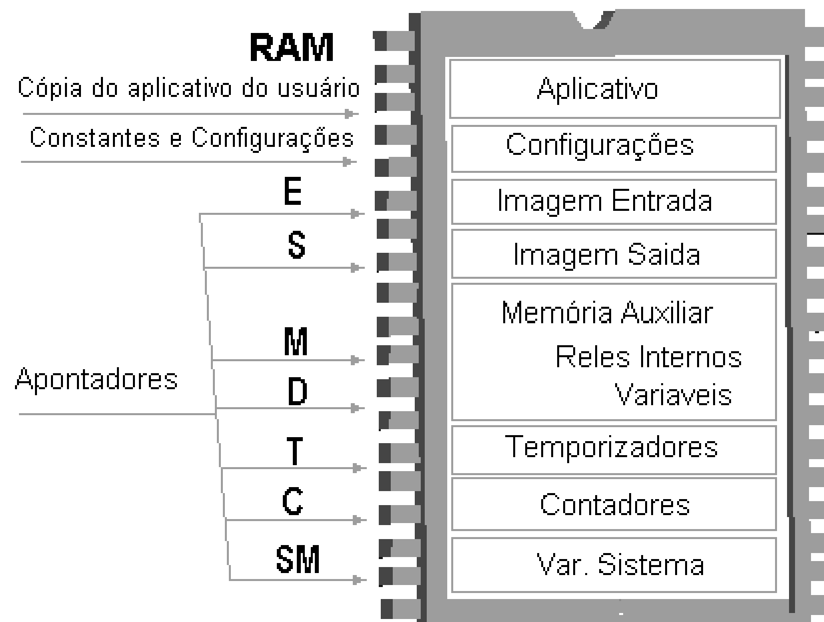


Figura 3-65 Organização interna memória RAM

Neste trabalho utilizamos operando genérico, utilizando a seguinte simbolização para referenciar os diversos operando de controlador:

- Ex – Endereçamento de uma entrada, ou seja, letra E seguida de um número seqüencial.
- Sx – Endereçamento de uma saída, a letra S seguida de um número sequencial.
- Mx – Endereçamento de um operando de memória auxiliar, a letra M seguida de um número.
- Dx – Endereçamento de um dado de 16 bits, a letra D seguida de um número.
- Cx – Contador, a letra C seguida de um número.
- Tx – Temporizador a letra T seguida de um número.

3.9 Programa

Podemos definir programa como um conjunto de atividades pré-estabelecidas escritas em conformidade com um padrão de linguagem, que impõe as ações à máquina.

Para exemplificar melhor, poderíamos considerar o ser humano como uma máquina e neste caso um programa simples desta “máquina” poderia ser representado pela sequência de atividades inseridas na tabela 3-7.

Atividade	Descrição
01	Levantar-se com a finalização do tempo de descanso
02	Tomar banho
03	Alimentar-se
04	Deslocar-se até trabalho/escola/universidade
05	Desenvolver atividades do trabalho/escola/universidade
06	Esperar meio-dia e almoçar
07	Continuar as atividades do período da manhã
08	Finalizar as atividades às 18 horas
09	Retornar para casa
10	Tomar Banho
11	Jantar ou Café
12	Atividade com a família/amigos/escola
13	Descansar
14	Inicializar novamente no passo 01

Tabela 3-8 Programa simples

Este pequeno programa corresponde à atividade em blocos da “máquina humana”. Cada bloco de atividade corresponde a uma função que pode ser subdividida em outras atividades, por exemplo, no passo 11 não definimos os movimentos das mãos, boca e outros para que o corpo possa receber o café! Em outro momento da descrição realizaremos esta particularização. Ainda para o exemplo citado será necessário alguns ajustes entre uma máquina e outra, como por exemplo, o tempo para intervalo de almoço e descanso.

O programa de um controlador é desenvolvido através de uma linguagem em que a máquina entenda sua codificação e ao mesmo tempo seja de fácil compressão por parte do programador, para que o processo de programação e manutenção ocorra com o menor tempo possível. A facilidade de programação também influência na flexibilização do programa para ajustes no conjunto de atividades programadas.

A primeira linguagem utilizada para programação das máquinas eletrônicas foi a linguagem de mnemônicos que consiste em fazer uma analogia direta da linguagem de máquina ou binária com códigos do alfabeto, no caso o alfabeto inglês. Na tabela 3-8 podemos ver um pequeno programa em mnemônico de movimentação e soma de dois valores.

Linguagem máquina (8085)	**Linguagem de Mnemônicos**	**Descrição**
78	Mov A,B	Movimenta valor de B para A
C6 08	Adi 08	Soma 08 com conteúdo de A e coloca o resultado em A
D3 20	Out 20	Envia resultado para o endereço 20
C9	Ret	Retorna ao inicio

Tabela 3-9 Linguagem de máquina

Para desenvolvimento de um programa mais sofisticado que exija um número maior de comandos, a linguagem mnemônica não é a ideal, pois precisaríamos de muito tempo para o desenvolvimento da aplicação. Para reduzir o tempo e facilitar a programação foram criadas novas linguagens, que tornaram mais legível o código elaborado. As linguagens mais fáceis de entender, que são mais legíveis, são classificadas como linguagens de alto nível, ou seja, baixo nível corresponde a linguagem de máquina, alto nível corresponde a nossa linguagem corrente, quanto mais próximo da nossa linguagem estiver a programação, mais alto nível será está linguagem.

Depois de criar o programa deve-se utilizar um compilador para converte a linguagens de alto nível para o código binário que será enviado para a memória do controlador. Os compiladores fazem parte dos ambientes de programação fornecidos pelos fabricantes dos controladores, e na maioria das vezes o processo de compilação é feito de forma automática, durante a transmissão e recepção do programa para o controlador.

Para programar controladores lógicos existem algumas linguagens padronizadas através da norma IEC 61131-3. Estas linguagens são disponibilizadas nos aplicativos de programação dos controladores de forma individuais e/ou em conjuntos, conforme interesse de cada fabricante.

As cinco linguagens definidas são:

- Lista de Instrução;
- Texto Estruturado;
- Ladder;
- Diagrama de Blocos;
- Diagrama Funcional Sequencial.

3.9.1 Programação: Lista de Instrução

Lista de Instrução, também conhecida por programação Booleana e Mnemônica, e uma linguagem alfanumérica que consiste numa linguagem de codificação direta da linguagem de máquina em abreviações pré-definidas dos comandos escritos em inglês.

Muitos fabricantes permitem a programação de seus controladores, também nesta linguagem, visto que o programa original de outra linguagem será convertido para a linguagem de máquina. Na tabela 3-9 temos um pequeno programa em lista de instrução que realiza a operação "e" entre dois operando.

Lista de Instrução	Descrição
STR E0	Carrega o valor de E0
AND E1	Realiza a operação AND entre E1 e E0
OUT S0	Envia resultado para S0
END	Fim de programa

Tabela 3-10 Lista de instrução

3.9.2 Programação: Texto Estruturado

A linguagem de texto estruturado é uma linguagem alfanumérica de alto nível, que apresenta uma descrição de atividades estruturadas através de comandos alfanuméricos em blocos. Esta linguagem comporta-se de forma semelhante às linguagens de desenvolvido para os computadores domésticos, assim como: linguagem C e pascal. Na tabela 3-10 podemos ver um exemplo desta linguagem, onde é realizada uma operação “e” entre dois operando.

Texto estruturado	Descrição
Valor := Entrada_0 AND Entrada_1	Realiza operação booleana
Saida_0 := Valor	Envia para a saída
END	Fim de programa

Tabela 3-11 Texto estruturado

3.9.3 Programação: Linguagem Ladder

A linguagem Ladder é uma linguagem gráfica, sendo a mais utilizada no ambiente industrial, pois a programação é feita através de símbolos que representam os circuitos elétricos de relê, por isso esta linguagem também é conhecida por linguagem de relês. A linguagem ladder é uma representação sofisticada de símbolos e blocos funcionais que permitem a manipulação de informações de um controlador lógico. Basicamente esta linguagem consiste da energização dos blocos funcionais para a realização de atividades. Esta linguagem será particularizada no próximo capítulo.

Inicialmente eram utilizados apenas os seguintes blocos:

- ➲ NA – contato Normalmente Aberto;
- ➲ NF – contato Normalmente Fechado;
- ➲ Bobina – dispositivo de saída.

A figura 3-23 é um exemplo da linguagem ladder, representando uma chave de partida direta. Para permitir a comparação com o sistema elétrico colocamos ao lado o diagrama elétrico da chave de partida direta.

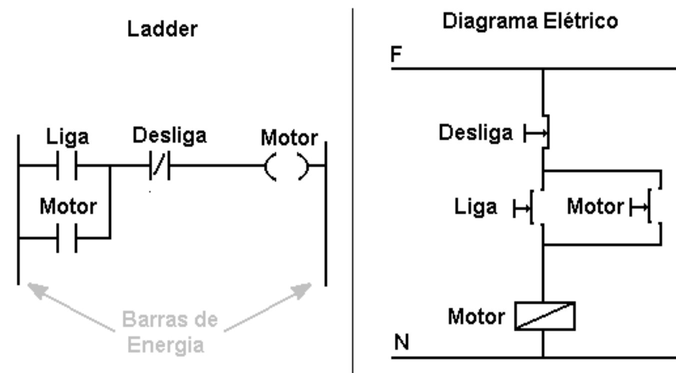


Figura 3-66 Diagrama ladder X elétrico

3.9.4 Programação: Diagrama de Blocos

Diagrama de blocos é uma linguagem gráfica também conhecida por diagrama de blocos de funções. A programação nesta linguagem consiste em representar graficamente as operações da lista de instrução, através da interligação dos blocos representativos de cada função.

No exemplo da figura 3-24, vemos uma lógica “e” entre E0 e E1, e um bloco “timer” para temporizar esta operação com resultado na saída S1. A entrada E3 serve para inicializar (reset) o temporizador.

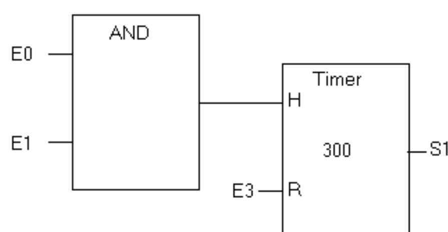


Figura 3-67 Diagrama em blocos de funções

A linguagem de diagrama de blocos é bastante utilizada hoje em dia pelos pequenos controladores, também chamados de NANOCLP ou reles temporizados.

3.9.5 Programação: Diagrama Funcional Sequencial

Diagrama funcional sequencial, ou SFC do inglês, *Sequential Function Chart*, é uma linguagem semelhante ao Grafcet, que permite descrever as

atividades sequenciais de uma máquina através de etapas, ações, transições e condições.

O SFC é dividido em etapa de execução, para cada etapa existe uma ou mais ações que serão executadas. Entre duas etapas existe uma transição, e para cada transição uma ou mais condições que farão o sistema evoluir de uma etapa a outra. A figura 3-25 corresponde a um exemplo de um SFC.

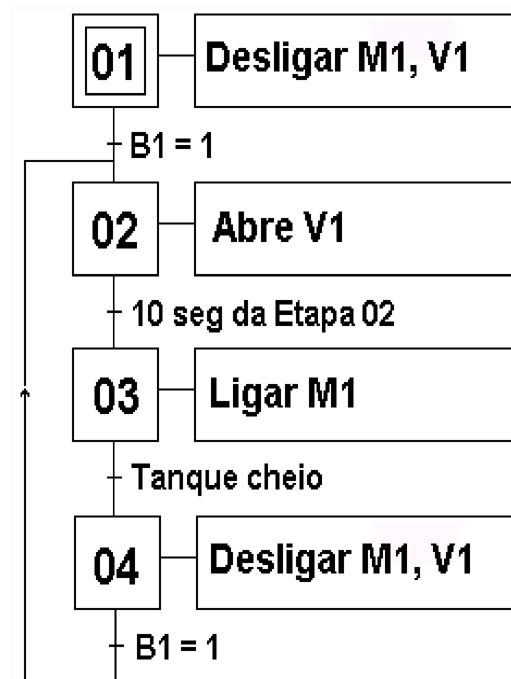


Figura 3-68 Diagrama SFC

No capitulo cinco veremos a utilização de diagrama de estados para a definição de sistemas de controle, o que facilita a programação em SFC.

4 - Programando em Ladder

"Quanto maior se foi tornando a solidariedade entre mente e mãos, tanto mais o suporte foi virando mundo e a vida, existência. O suporte veio fazendo-se mundo e a vida, existência, na proporção que o corpo humano vira corpo consciente, captador, apreendedor, transformador, criador de beleza e não 'espaço' vazio a ser enchido por conteúdos."

FREIRE, 1996, p.57

Estudaremos neste capítulo a linguagem de programação Ladder, abordando a forma de programação e os principais comandos. Para facilitar a compreensão dividimos o capítulo em duas partes principais como: funções básicas e funções avançadas.

4.1 Definição

A linguagem ladder é uma representação gráfica, similar ao desenho de um diagrama elétrico. O ladder é baseado em duas barras de energia virtual, verticais situadas à esquerda e direita do ambiente de programação. A programação consiste em criar uma corrente elétrica virtual através de símbolos gráficos inseridos entre estas duas barras.

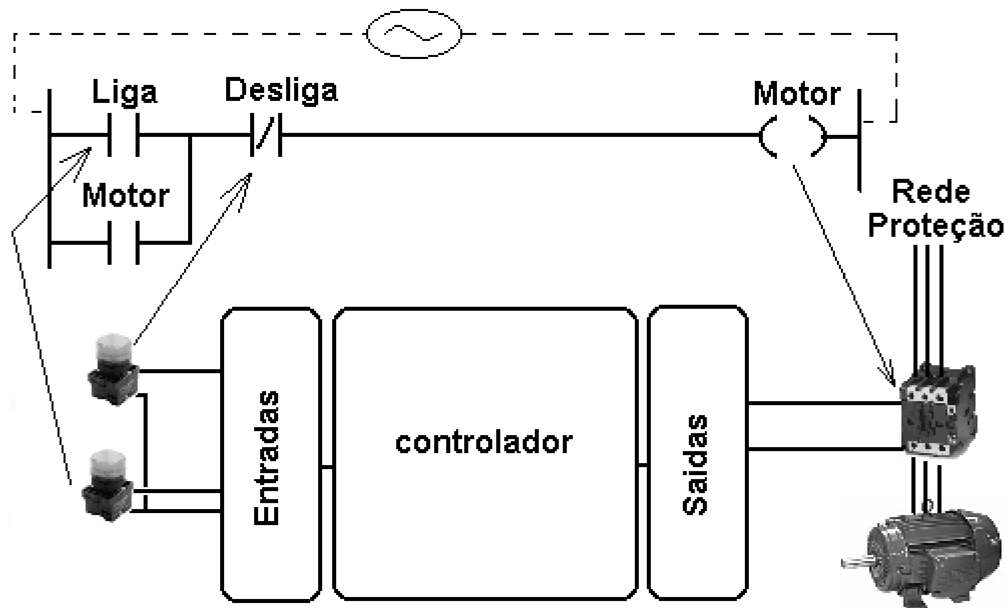


Figura 4-69 Diagrama ladder

4.1.1 Regras Básicas de Programação

A programação em ladder é feita com base nas seguintes regras básicas:

- A linguagem Ladder é constituída por duas barras de energia virtual, uma à direita e outra à esquerda.

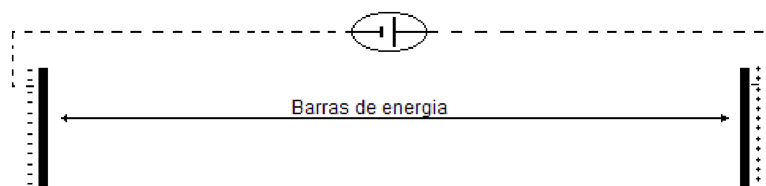


Figura 4-70 Barras de energia virtual

- A programação consiste em criar uma corrente elétrica virtual que deve circular entre estas duas barras, sempre no sentido da esquerda para direita.

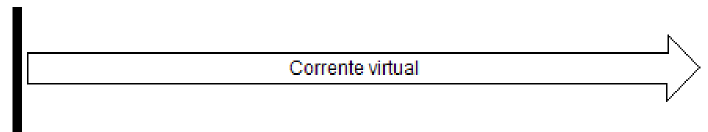


Figura 4-71 Corrente elétrica virtual

- Para controlar a corrente elétrica devemos utilizar elementos de programação de entrada, que tem a função de bloquear ou liberar esta corrente.

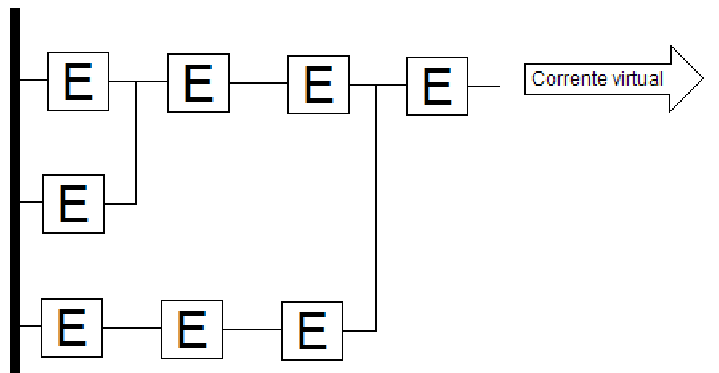


Figura 4-72 Elementos de entrada

- Os elementos de entrada somente podem ser utilizados entre a primeira coluna da lógica até a penúltima coluna. Estes elementos podem ser associados em série e paralelo nos limites das lógicas.
- A corrente virtual controla os elementos de saída, os elementos de saídas alteram os seus estados conforme a existência ou não de uma corrente virtual.
- Os elementos de saídas somente podem ser utilizados na última coluna da lógica e podem ser associados em paralelo.

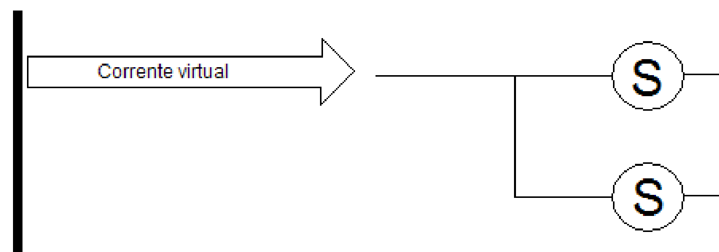


Figura 4-73 Elementos de saída

4.1.2 Endereçamento de Variáveis

O endereçamento de variáveis corresponde ao acesso às posições de memórias do controlador. Cada endereço utilizado corresponde a uma região de memória ou a um dispositivo de entrada ou saída do circuito do controlador lógico. As entradas e saídas digitais e analógicas, os registradores, temporizadores e contadores são operandos do controlador lógico que podem ser utilizados em um programa em ladder. Para utilizar-se destes operandos é necessário criar uma referência ou endereçamento para cada um deles, através de apontadores para cada região de memória.

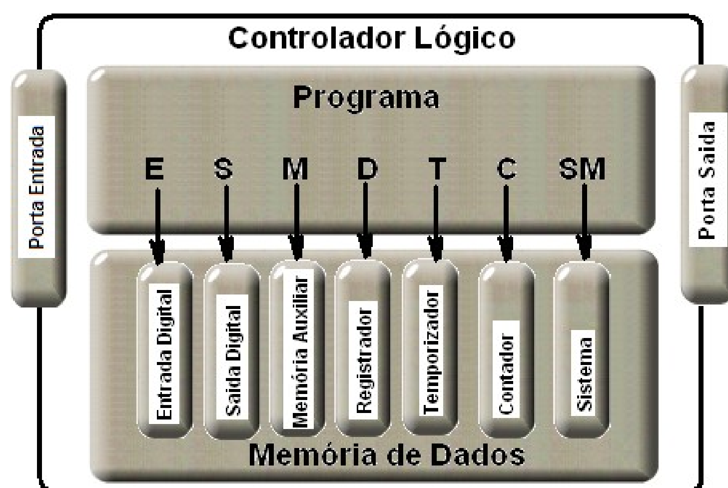


Figura 4-74 Endereços do sistema

4.1.3 Lógicas de Programação

É uma fração de um programa correspondente a um determinado número limitado de instruções, que define a ordem de execução das instruções nesta parte do programa.

O programa será executado lógica por lógica, a cada ciclo de varredura iniciado na primeira lógica do programa até a última lógica, quando o sistema reinicia o processo de varredura.

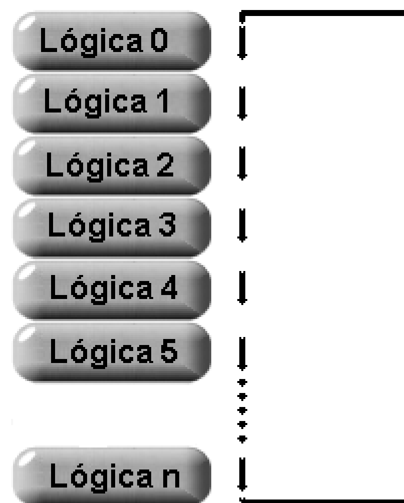


Figura 4-75 Sequência de execução das lógicas

Na lógica, o processamento das instruções ocorre dentro de uma ordem definida pelo próprio fabricante do controlador lógico, existindo duas formas básicas de controle de processamento de lógicas, a primeira mais simples, consiste na execução por coluna, onde o processamento se dá de coluna a coluna, iniciando-se pela coluna mais à esquerda até a atingir a última coluna à direita (Figura 4-8A). O outro método de controle de lógicas consiste na execução da lógica linha e coluna, ou seja, executa-se a lógica por linha e armazena-se a condição lógica das ligações verticais encontradas na linha para retornar e continuar a execução das linhas a partir desta condição (Figura 4-8B). Para evitar erros de programação é importante que se tenha o conhecimento prévio deste processamento, através dos manuais de programação dos controladores lógicos.

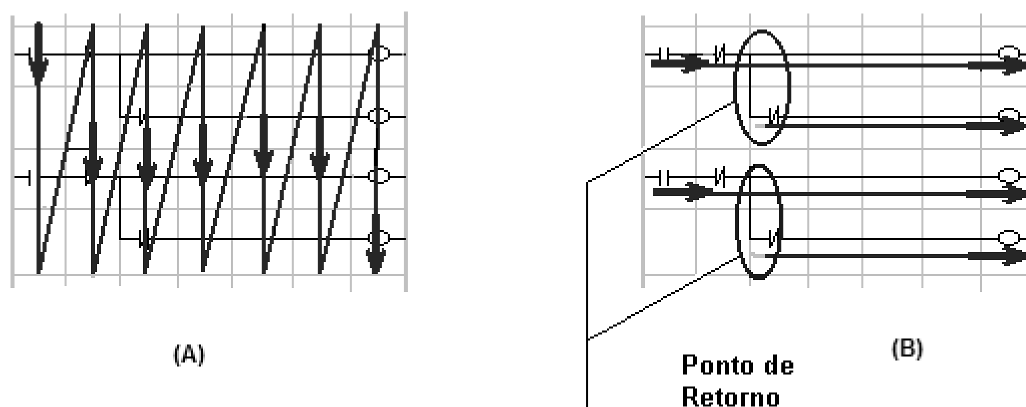


Figura 4-76 Sequência de execução interna das lógicas

Na figura 4-8 podem-se visualizar os dois principais métodos utilizados pelos controladores lógicos para execução do programa. No primeiro método o controlador executará a lógica por colunas, enquanto que no segundo o método a sequência é feita por linha, sendo que os pontos de derivações verticais são guardados, a fim de continuar está ramificação ao final da linha atual.

Veja o que acontece com a lógica da figura 4-9, sendo executada em controladores com tecnologias diferentes. No CP de tecnologia por coluna, temos um circuito multivibrador astável, que gerará um pulso a cada troca de sinal de E01, enquanto no segundo método, a saída permanecerá sempre desligada, e para gerar o pulso como na tecnologia anterior precisaríamos modificar a lógica.

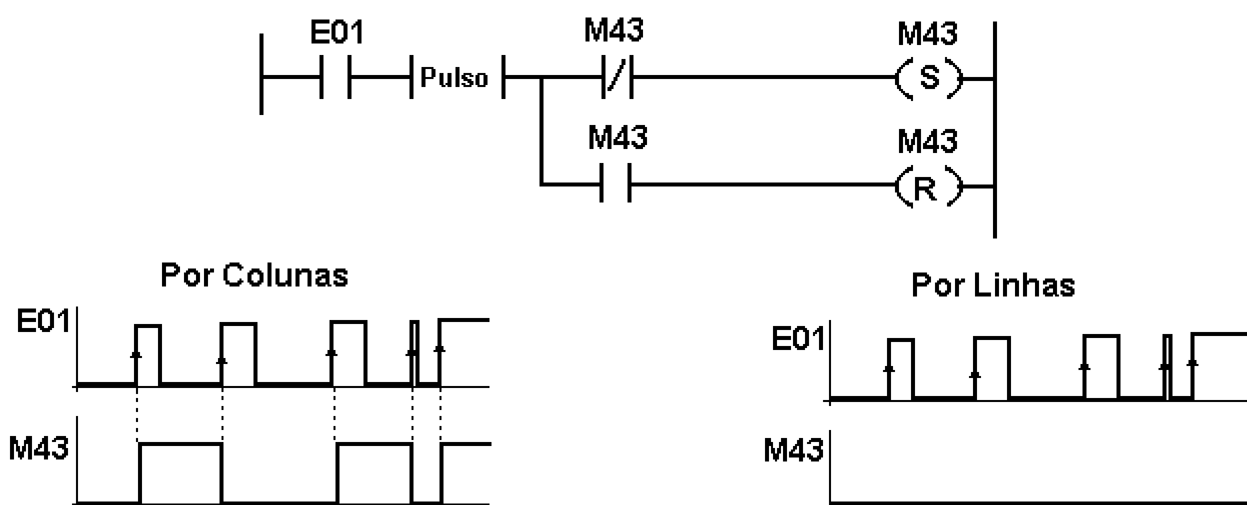


Figura 4-77 Exemplo execução interna de lógica

4.2 Contatos NA, NF e Bobinas

Os elementos básicos de programação em ladder são divididos em dois grupos, os elementos de programação de entrada e os elementos de saída.

4.2.1 Elementos de Entrada

Os elementos de entrada servem para controlar o fluxo da corrente elétrica virtual, permitindo bloquear ou liberar a passagem desta corrente. Os contatos NA (normalmente aberto) e NF (normalmente fechado) são os dois elementos de entrada básicos de programação.

O contato NA representado na figura 4-10, é um elemento que faz referência a um operando binário do sistema. Este operando corresponde a um endereço de entrada, saída ou mesmo um operando de memória auxiliar.

Sempre que o valor do operando estiver com nível lógico um, o contato NA deixará fluir a corrente virtual de sua entrada (lado esquerdo) para a sua saída (lado direito).

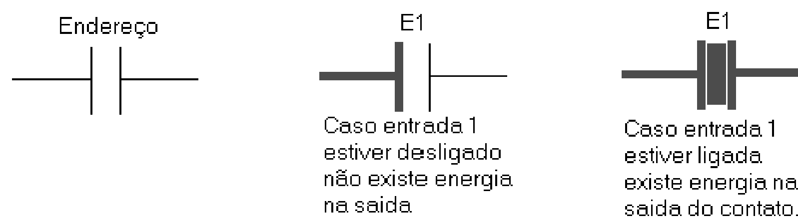


Figura 4-78 Contato NA

O contato NF, também é um operando binário, com endereçamento semelhante ao contato NA. Quando o elemento que referência este contato estiver com nível um, o elemento bloqueia a corrente virtual. Quando o endereço que referencia o contato estiver com nível lógico zero, o elemento permite a passagem da corrente virtual entre seus terminais. O contato NF esta representado na figura 4-11.

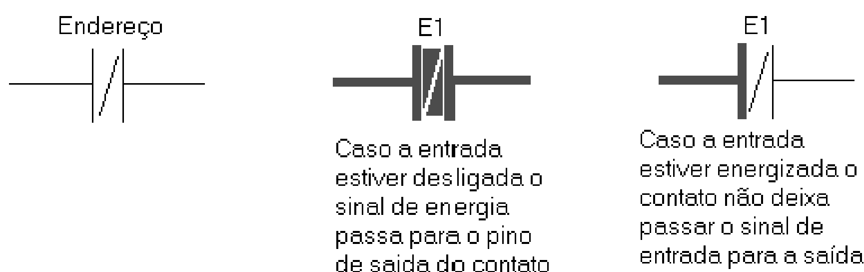


Figura 4-79 Contato NF

4.2.2 Elemento de Saída

O rele bobina é um elemento de programação que também referencia operandos binários. O rele bobina é utilizado para finalizar uma linha de programação, pois ele constitui um elemento de saída. Este elemento pode assumir endereços de qualquer operando binário, tais como: saída e operandos de memórias auxiliares. Nunca utilizaremos os operandos de entradas para referenciar os elementos de saída, pois estes sempre serão um espelho das portas de entradas físicas de nosso sistema.

O rele bobina é um objeto de programação que coloca o valor um no conteúdo do endereço do elemento caso exista corrente virtual na entrada do elemento, caso contrario coloca o valor a zero.

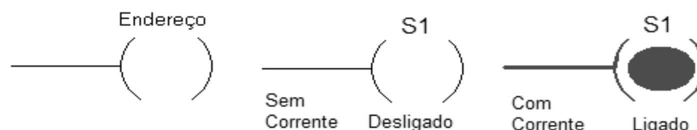


Figura 4-80 Rele bobina

4.2.3 Ladder : Lógicas Básicas

Na figura 4-13 exemplificamos a utilização de contatos NA, NF e rele bobina, onde vemos os três elementos conectados em série, sendo que o rele bobina é posicionado mais à direita, como elemento de saída. O rele bobina endereçado por S01 ativa, quando o contato NA com E01, estiver com nível UM e o contato NF de E02, com nível ZERO.

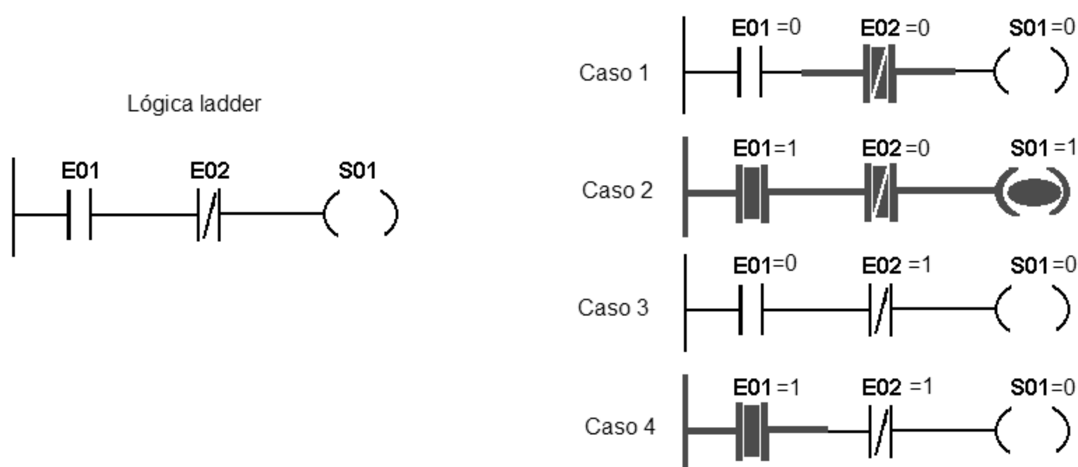


Figura 4-81 Função lógica E em ladder

No segundo exemplo, o rele bobina, endereçado por S03 fica ativo quando o conteúdo do endereço E04 for ZERO, acionando o contato NF. Também aciona quando o conteúdo de E07 for UM, pois aciona o contato NA. O conteúdo de S03 é igual a UM quando E04 igual a ZERO OU E07 igual a UM.

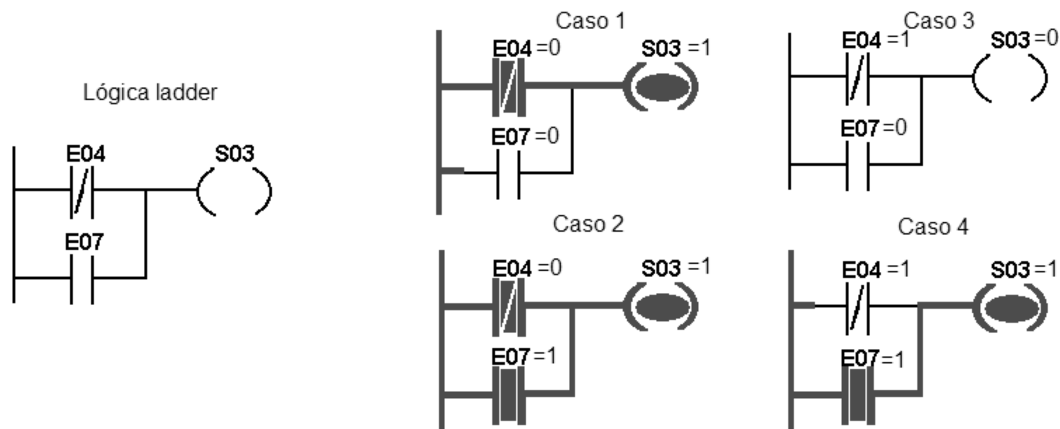


Figura 4-82 Função lógica OU em ladder

A figura 4-13 faz referencia a uma operação AND e a figura 4-14 faz referencia a uma operação OR, e este fato é muito útil na simplificação de lógicas, através de otimização utilizada na eletrônica digital, como os mapas de Karnaugth, para reduzir o tamanho de lógicas.

4.3 Exemplos com Contatos e Bobinas

4.3.1 Partida direta de motor

Representar uma chave de partida direta com a utilização de lógicas ladder. Sabendo-se que o circuito elétrico a ser utilizado nesta aplicação é o da figura 4-15 e a legenda da figura é:

- B1 – botoeira liga (NA),
- B0 – botoeira desliga (NF),
- Em1 – Chave de emergência (NA),
- k1 – Contator que liga o circuito de força para comando do motor.

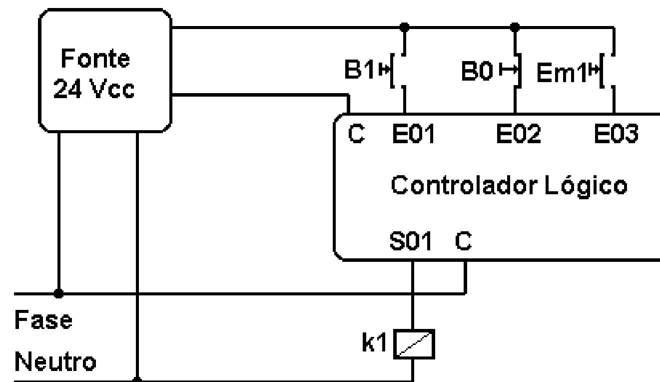


Figura 4-83 Exemplo 1: esquema elétrico

A solução em ladder é indicada na figura 4-16. Observamos que o motor liga quando a saída S01 energiza isto ocorre quando E01 for UM, ou seja, pulsando-se a botoeira B1 e mantendo a botoeira B0 em repouso. B0 em repouso mantêm a entrada E02 habilitada, pois seu contato é NF. O botão de emergência também deve ficar em repouso para habilitar a entrada E03. Na lógica ladder o contato NA de S01 faz a retenção do comando, pois E01 é apenas um pulso. Para desligar deve-se acionar E02 ou E03, cortando a corrente do circuito.

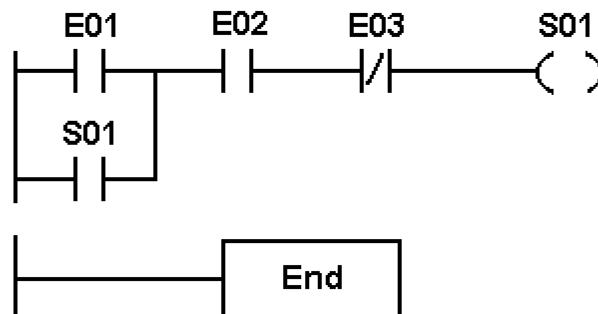


Figura 4-84 Ladder: chave de partida direta

4.3.2 Controle de Agitador

Desenvolver um algoritmo em ladder para controle de um agitador, sabendo-se que o agitador será ligado sempre que a manopla de comando (H) estiver na posição de liberação e o nível do tanque (S01) acima do sensor de nível utilizado (contato fechado).

O motor ligará através do comando da manopla e do sensor, ou seja, a manopla e sensor serão representados por contatos NA e o motor por um rele bobina, todos dispostos em serie, como mostra o diagrama da figura 4-18. Observe que o motor aciona quando a saída do controlador for energizada devido ao comando (corrente virtual) imposta pelos contatos NA de E01 e E02 estarem ativos.

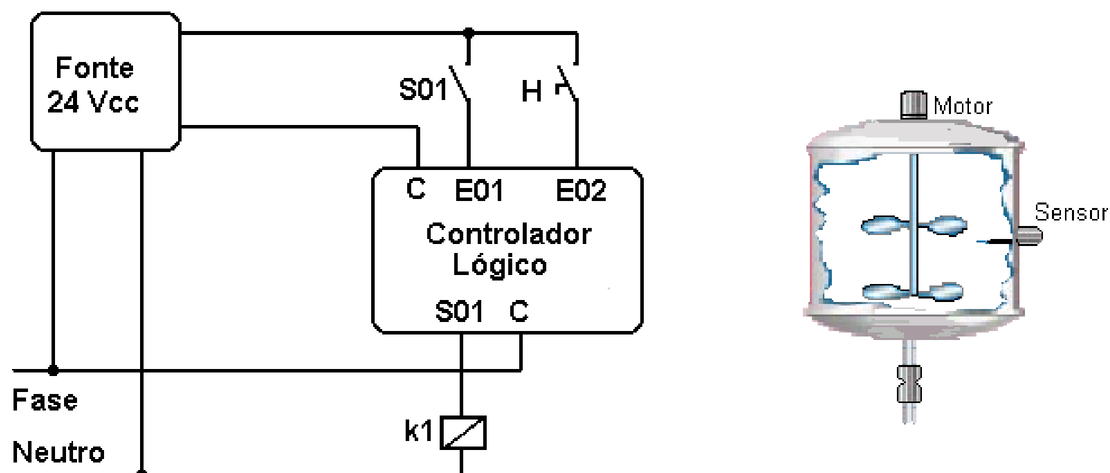


Figura 4-85 Controle de agitador (diagrama elétrico e fluxograma da fábrica)

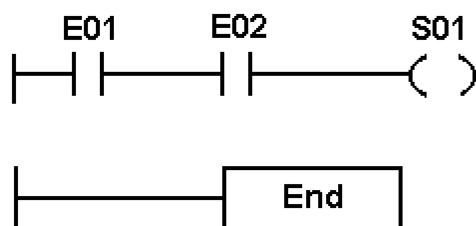


Figura 4-86 Solução: controle de agitador

4.3.3 Controle de Alarmes

Em um sistema automático uma característica muito importante relaciona-se com o retorno do estado dos equipamentos e a utilização de alarmes para sinalizar as condições inesperadas e eventos do processo. O sistema de alarme é dividido em duas partes uma sonora e uma apenas visual. Normalmente o alarme visual tem a função de sinalizar um alarme existente no sistema, permanecendo com a sinalização ligada até que cesse o evento que gerou o alarme. Já o alarme sonoro tem a função de alertar o operador para a ocorrência de um novo alarme ou um alarme critico em atividade. Para os alarmes críticos, como por exemplo, pressão alta na caldeira, este alarme

somente deixará de sinalizar quando finalizar o evento que o gerou, semelhante ao alarme luminoso. Quando o alarme gerado não for crítico ao sistema, o operador aciona um botão no painel para silenciar o alarme, indicando que já tomou ciência do ocorrido. O alarme sonoro dispara novamente quando surgir um novo alarme.

Para frisar este conceito sobre alarmes desenvolveremos a lógica de controle deste sistema, seguindo os procedimentos abaixo:

- O alarme luminoso deverá permanecer ligado enquanto existir um alarme ativo.
- O alarme sonoro será acionado sempre que ocorrer uma transição de zero para um em qualquer uma das entradas de alarmes.
- Depois de ligado o alarme sonoro permanecerá ativo até que seja pressionado o botão para silenciar o alarme.

As entradas e saídas do controlador estão especificadas abaixo:

- E01: Alarme – Rele térmico do motor 1.
- E02: Alarme – Rele térmico do motor 2.
- E03: Alarme – Botão de emergência pressionado.
- E04: Alarme – Nível alto no tanque.
- E05: Cala Alarme – Botão para silenciar o alarme.
- S01: Alarme sonoro.
- S02: Alarme luminoso.

A figura 4-19 representa a solução encontrada para este exercício. Os operandos de memórias auxiliares M01 a M04 são utilizados para memorizar que o alarme já foi reconhecido desabilitando o alarme sonoro. Na ocorrência de outro alarme o alarme sonoro soará novamente.

4.4 Exercícios Propostos

4.4.1 Controle de Nível

Criar um programa em ladder para controle de nível indicado na figura 3-11. Utilizando-se o procedimento a seguir:

- ➲ Abrir a válvula de entrada e ligar a bomba, quando o sensor de nível baixo do tanque estiver com sinal em zero;
- ➲ Desligar o motor e fechar a válvula com o nível do tanque estiver alto, ou seja, o sensor de nível alto esteja com sinal igual a um;
- ➲ Em qualquer momento, quando a chave LIGA estiver desligada, o sistema permanece bloqueado, ou seja, o motor desligado e a válvula fechada;

O controle dos dispositivos de saída não faz parte do escopo de controle deste procedimento.

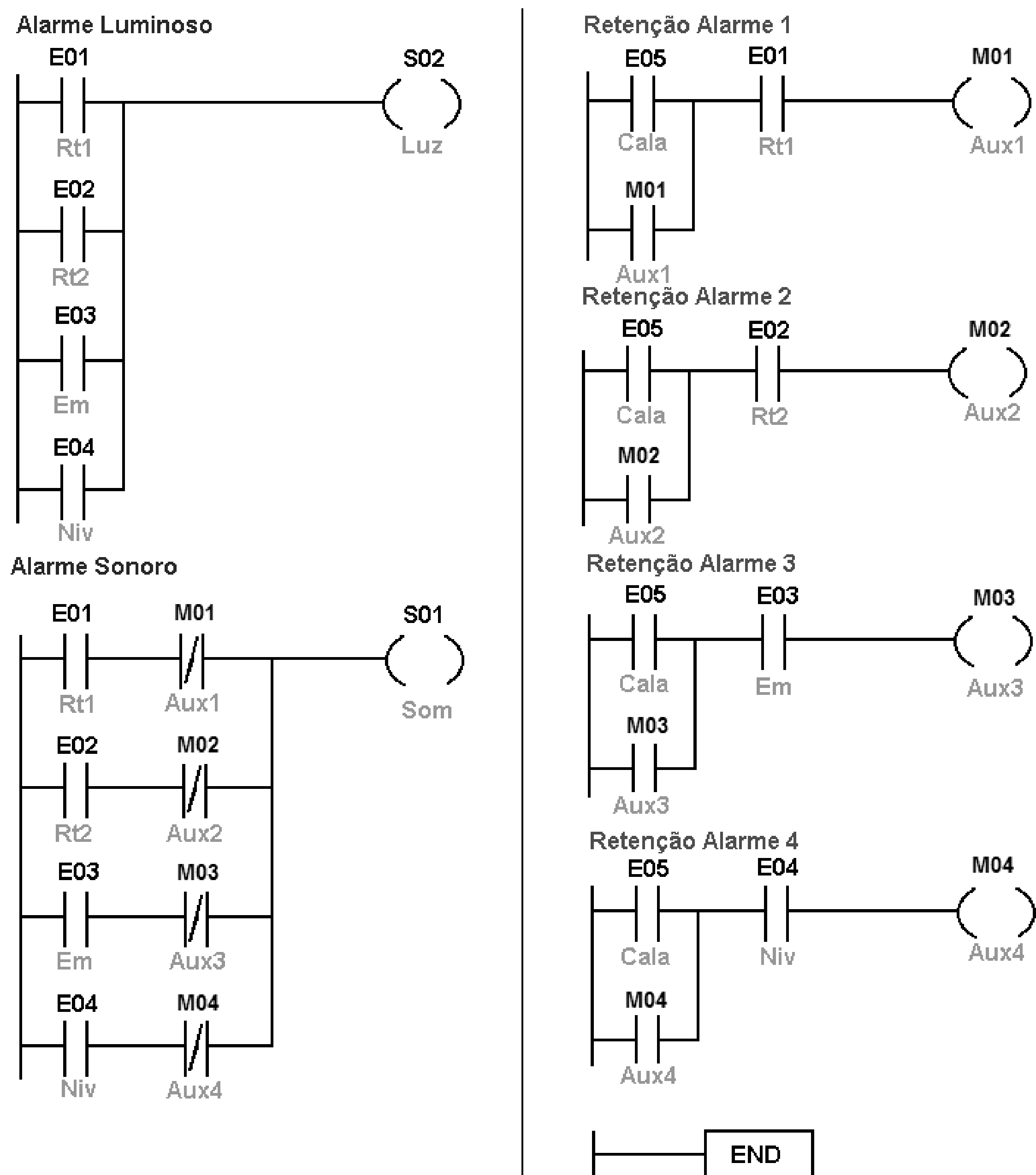


Figura 4-87 Solução controle de alarmes

4.4.2 Partida de motores

Controlar a partida de dois motores com funções diferentes através de uma única chave eletrônica de partida, onde temos:

Em: Emergência Habilita Geral;

B1: Liga Motor 1;

B2: Liga Motor 2;

B3: Desliga Motor 1;

B4: Desliga Motor 2;

COK: Retorno Chave Eletrônica Ligada;

C100: Retorno Chave Eletrônica a 100% da tensão da rede, ou seja, trocar alimentação;

K1: Liga Motor 1 diretamente da rede;

K2: Liga Motor 1 via chave eletrônica;

K3: Liga Motor 2 diretamente da rede;

K4: Liga Motor 2 via chave eletrônica.

O comando inicia através do botão B1 ou B2 que aciona o contator k2 ou k4 para a partida via chave eletrônica, ao retornar o sinal C100 o controlador deverá chavear o contator k1 ou k3, desligando k2 ou k4. Os contatores k2 e k4 não devem ser ligados ao mesmo tempo e somente são acionados quando o botão de emergência e o sinal COK estiverem liberados. O botão B3 desliga k1 e k2 e o botão B4 desliga k3 e k4.

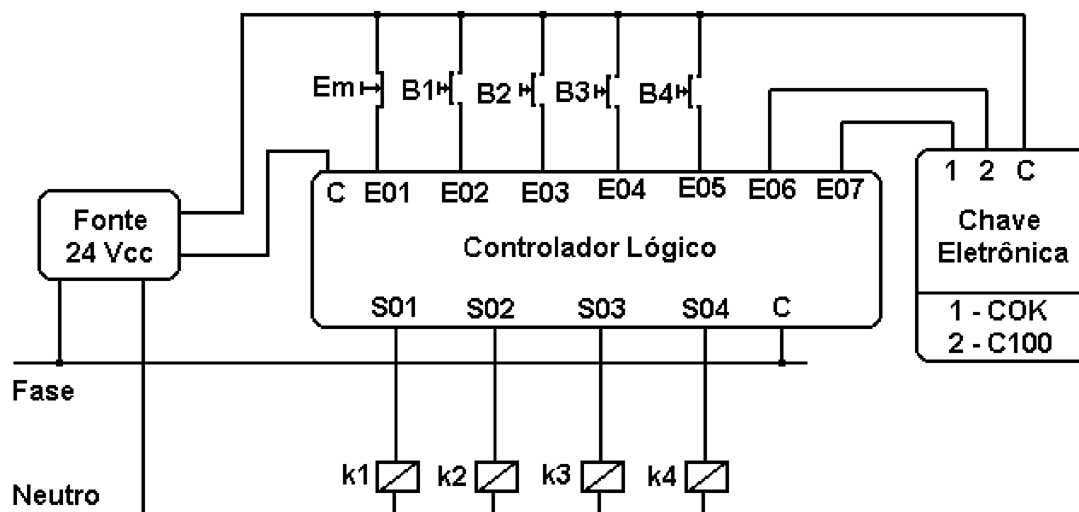


Figura 4-88 Esquema elétrico, partida eletrônica de dois motores

4.4.3 Transporte de Matérias Primas

Controlar o acionamento de um transportador de correia, representado na figura 4-21, a fim de manter o depósito de utilização e pré-aquecimento sempre cheio. Neste exercício temos:

Em: Emergência Habilita Geral;

MA: Manopla manual (0) e automático (1);

B1: Liga Correia 1 no modo manual;

B2: liga Correia 2 no modo manual;

LL1: Silo principal vazio (desabilitar correia);

LH1: Silo principal cheio (habilitar correia);

LL2: Nível baixo tanque depósito (ligar transportador);

LH2: Nível alto, carga completa tanque depósito (desligar transportador);

k1: Liga Correia1 (M1);

k2: Liga Correia 2 (M3);

k3: Liga Elevador (M2).

Observação: os sensores utilizados são do tipo pás móveis, ou seja, retornam um sinal igual a um (1), quando suas pás são impedidas de movimentarem, isto ocorre quando tivermos produtos junto às pás destes sensores.

A habilitação das correias acontece quando atingir o nível alto no tanque principal (LH1) e/ou quando o controlador é energizado pela primeira vez.

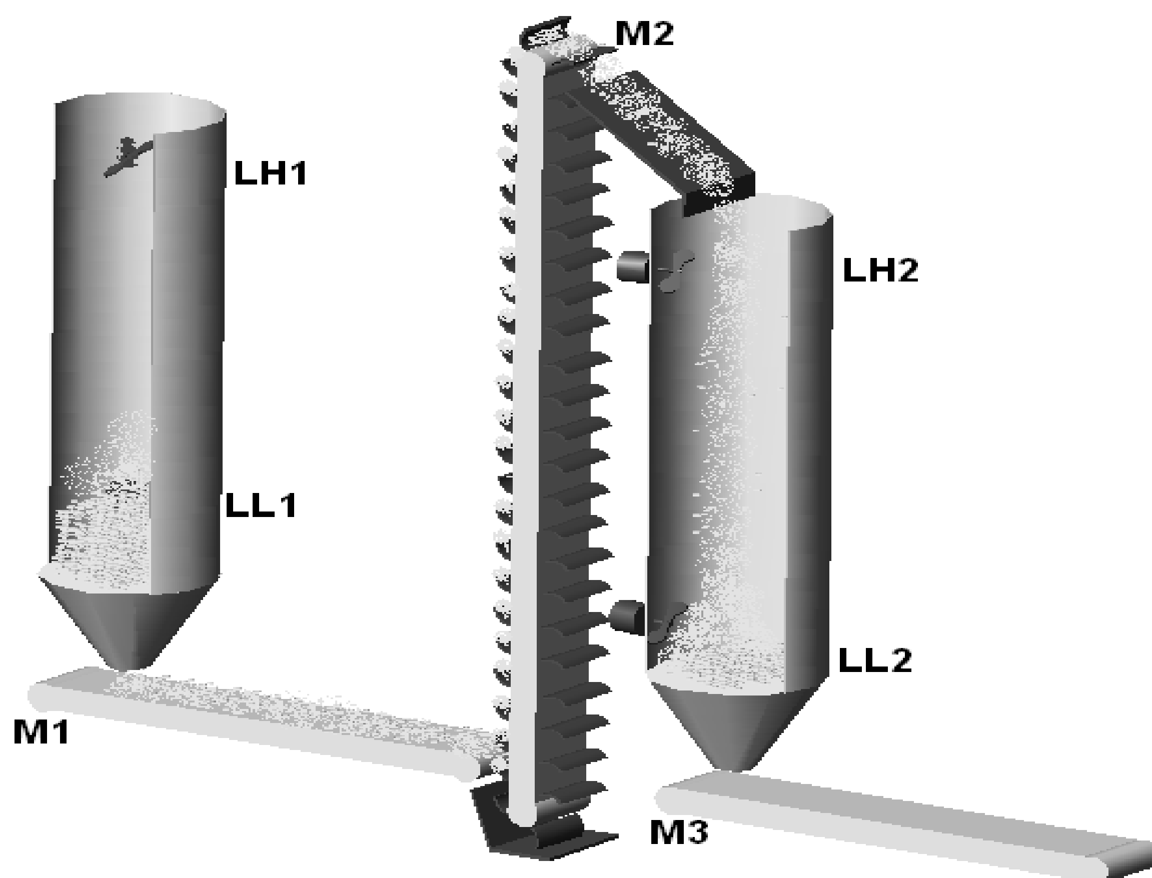


Figura 4-89 Fluxograma: transporte de matérias primas

O circuito elétrico deste exercício é mostrado no diagrama da figura 4-22.

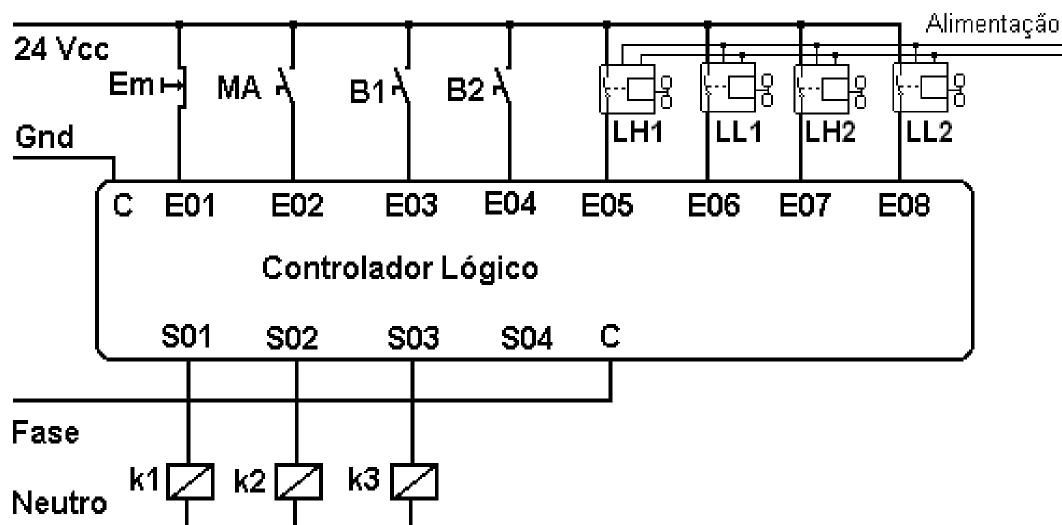


Figura 4-90 Circuito elétrico : sistema de transporte

4.4.4 Sistema de dosagem

Em um sistema de dosagens de macro-ingredientes, controlar o deslocamento de um carro de mistura de forma automática, através das seguintes condições:

- ➲ Ligar o carro para direita sempre que for pressionada a botoeira B0, ou seja, B0 = 1;
- ➲ Parar o movimento nas posições 1, 2 e 3;
- ➲ Retornar o carro para posição 1 ao chegar ao fim de curso que indica fim de trilho.

Observação à descarga do produto no carro é feita de forma manual, após o carro parar na posição, o operador abre a comporta de dosagem, colocando a quantidade desejada do produto no carro, e ao final pressiona a botoeira B0, indicando que o carro deve avançar para a próxima posição.

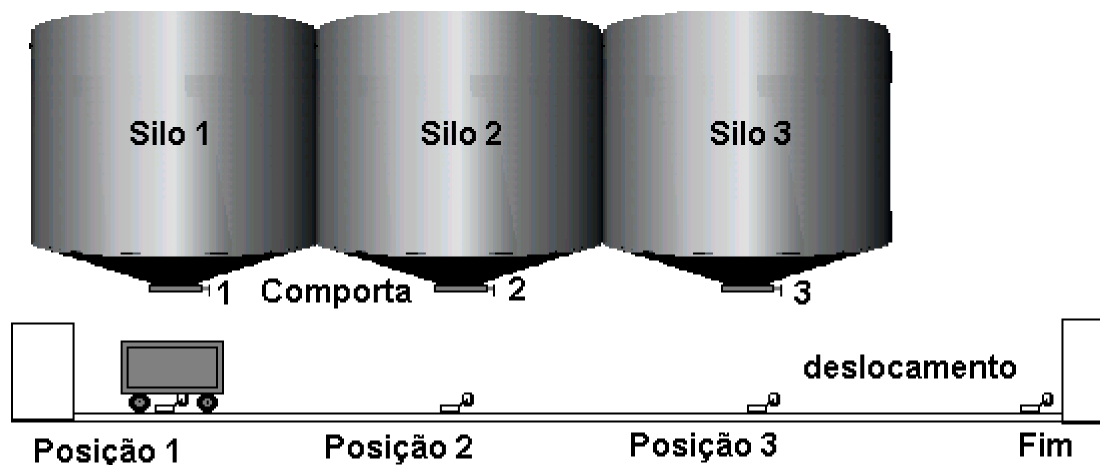


Figura 4-91 Sistema de dosagem

4.5 Contadores e Temporizadores

Os contadores e os temporizadores são objetos de programação utilizados para auxiliar no controle de dispositivos.

4.5.1 Contadores

O contador é um elemento de programação utilizado para acumular eventos que ocorrem no sistema. É comumente utilizado em lógicas com as seguintes finalidades:

- Contagem de peças;
- Auxiliar na manipulação de menus em IHM;
- Auxiliar na temporização de minutos e horas.

Um evento somente é acumulado no contador quando ocorre uma transição na entrada de contagem. O contador exige a indicação dos seguintes operandos para o seu funcionamento:

- Operando binário para entrada de contagem;
- Operando binário para entrada de reset;
- Registrador 16 bits para acumulação dos eventos (PV);
- Registrador de 16 bits para valor de referência (SP);
- Operando binário de saída do contador.

A entrada habilita determina que o valor do contador seja incrementado do valor um, ou seja: CONTADOR = CONTADOR + 1.

A entrada reset faz a inicialização do contador, ou seja, torna o seu valor igual a zero.

A saída do contador ativa sempre que as duas variáveis de 16 bits tiverem seus valores iguais (SP = PV). A declaração da saída depende da configuração interna do CP, existindo a forma explícita e a implícita. Na forma explícita o operando de saída no bloco do contador deve ser declarado. Na forma implícita o valor do operando de saída está relacionado ao endereço do acumulador, neste tipo de declaração temos endereços de memórias exclusivos para os contadores, sendo que o mesmo nome faz referência a um endereço binário de estado do contador e um registrador de 16 bits com o valor de contagem. Normalmente neste último caso o contador tem o endereço inicializado pela letra "C" seguido de um endereço numérico, este endereço faz referência ao valor do acumulador e ao operando de saída.

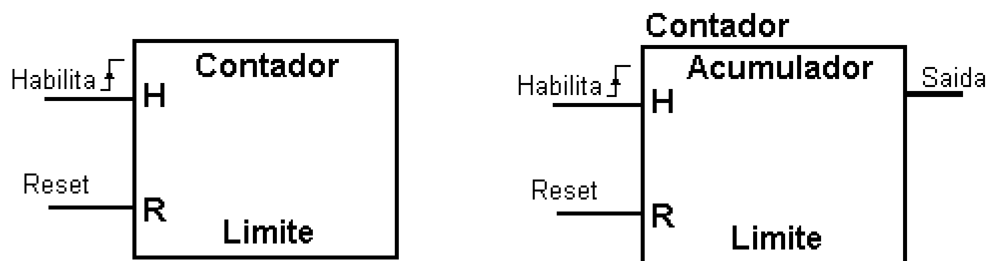


Figura 4-92 Simbolização contador (implícito e explícito)

4.5.2 Temporizadores

O temporizador é um elemento de programação utilizado para contagem de unidades de tempos, auxiliando no controle e monitoração de variáveis externas. A unidade de tempo mínima é chamada de base de tempo do temporizador e varia conforme o fabricante do CP, normalmente a base de tempo é dada em décimos de segundos.

Este elemento é comumente utilizado em lógicas com as seguintes finalidades:

- Atrasos de tempo entre dois equipamentos;
- Acumulação de tempo de processo;

- ➲ Circuito alternador (astáveis) para piscar alarmes;
- ➲ Histerese em controle de agitadores e níveis.

Existem dois tipos de temporizadores quanto a sua forma de inicialização, ou seja, com reset independente ou não. Abaixo temos a representação destes temporizadores, no temporizador sem o reset independente o sinal que habilita a temporização corresponde também ao sinal de inicialização, ou seja, o temporizador está habilitado temporizando algum evento, ou está com o seu valor em zero. Já no temporizador com reset independente, os sinais de habilitação e reset são independentes, e a temporização é feita de forma a acumular o tempo de habilitação deste elemento, a inicialização do temporizador ocorrer com a energização da entrada de reset, caso a entrada de habilitação e reset estiverem desabilitadas o temporizador permanece com o seu valor anterior.

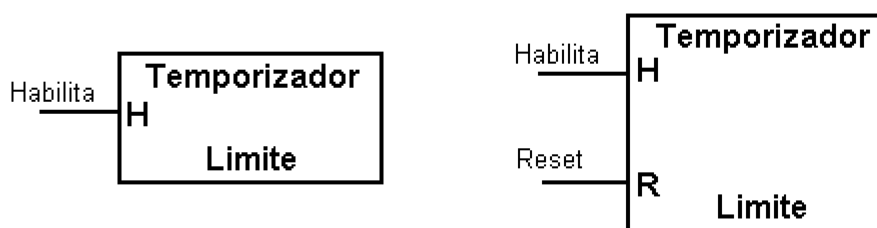


Figura 4-93 Representação de temporizador declaração implícita

O temporizador exige a indicação dos seguintes operandos para o seu funcionamento:

- ➲ Operando binário para habilitação;
- ➲ Operando binário para reset;
- ➲ Registrador 16 bits para acumulação do tempo (PV);
- ➲ Registrador de 16 bits para valor de referência (SP);
- ➲ Operando binário de saída do contador.

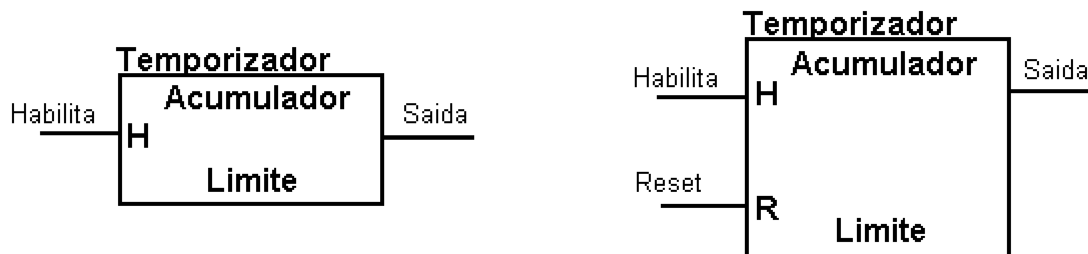


Figura 4-94 Representação de temporizador declaração explícita

Sempre que o tempo totalizado for igual ao valor de referência o temporizador (PV = SP) energiza o operando de saída. Dependendo do fabricante existem dois tipos de declaração do operando de saída. A primeira forma é a forma explícita, onde é declarado um operando de saída no bloco do temporizador. A outra forma é a implícita, ou seja, o valor do operando de saída está relacionado ao endereço do acumulador de tempo, neste tipo de declaração temos endereços de memórias exclusivos para os temporizadores, sendo que o mesmo nome faz referência a um endereço binário e um registrador de 16 bits. Normalmente neste último caso o temporizador tem o endereço inicializado pela letra "T" seguido do número de seu endereço, este endereço faz referência ao valor do tempo acumulado e ao operando binário de saída que indica o status do temporizador.

4.5.3 Exemplo 1: Partida Estrela-Triângulo de Motor

Representar uma chave de partida estrela triângulo com a utilização de lógicas ladder. Sabendo-se que o circuito elétrico a ser utilizado nesta aplicação é indicado na figura 4-27.

Onde:

Em: Emergência,

B0: Botoeira para desligar (NF),

B1: Botoeira para ligar (NA),

K1: Contator principal,

K2: Contator auxiliar conexão em triângulo,

K3: Contator auxiliar conexão em estrela.

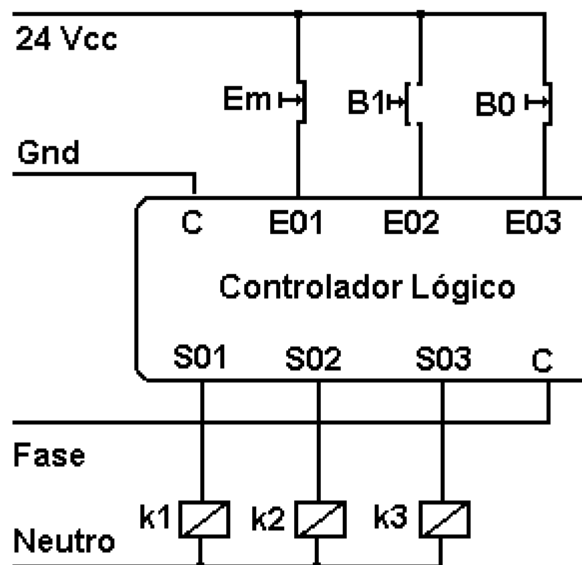


Figura 4-95 Diagrama elétrico controle estrela triângulo

O circuito elétrico original da chave estrela triângulo é apresentado na figura 4-28.

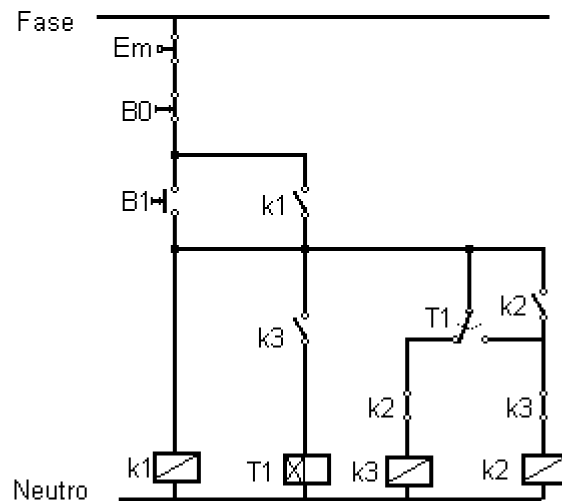


Figura 4-96 Chave estrela-triângulo

A figura 4-29 representa a solução deste exercício. A função do temporizador (T02) de 0,1 segundos é determinar o intervalo de tempo entre os contatores K2 e K3. O tempo de conexão em estrela é de 15 segundos e é determinado por T01.

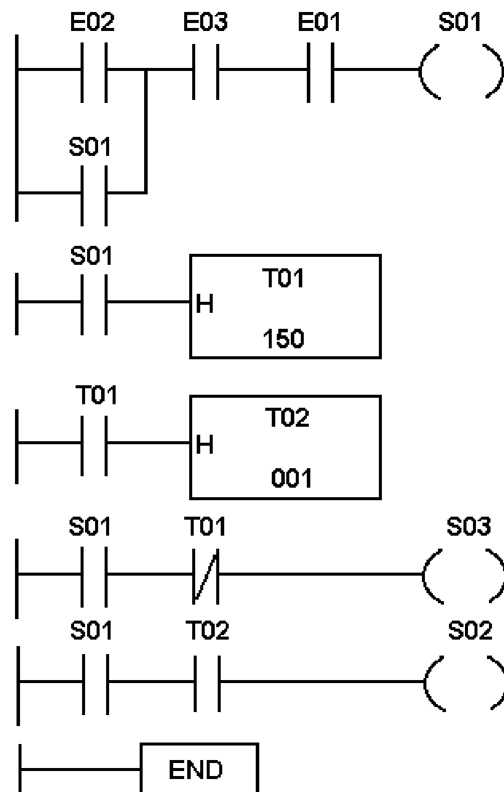


Figura 4-97 Solução ladder estrela-triângulo

4.5.4 Exemplo 2: Temporização Tempo de Falha

Verificar o tempo em minutos que um sistema fica em alarme. Neste exercício temos:

Entrada E01: Al1 (Alarme 01),
Entrada E02: Al2 (Alarme 02),
Entrada E03: Al3 (Alarme 03),
Entrada E04: inicializa sistema,
Saída S01: Exa (Lâmpada excesso de tempo maior que 120 minutos).

A solução do exercício proposto passa primeiro pela definição de um sinal comum para o alarme, através de lógica “ou” entre Al1 a Al3, colocando o sinal no auxiliar M1. Através do auxiliar M1 será habilitado um temporizador de 600 décimos de segundos (igual a um minuto). A variável de saída do temporizador será utilizada para resetar o próprio temporizador e incrementar um contador. O valor limite do contador será 120, ou seja, 120 minutos. O

contator ao atingir o limite (PV=SP) liga a lâmpada de excesso de tempo de alarme.

Na solução do exercício (figura 4-30) observamos que o contador incrementará através do sinal de saída do temporizador. Para resetar o temporizador será utilizado o sinal de saída do próprio temporizador e também o sinal de inicialização do sistema (E04), realizado através da lógica “ou” com resultado em M2.

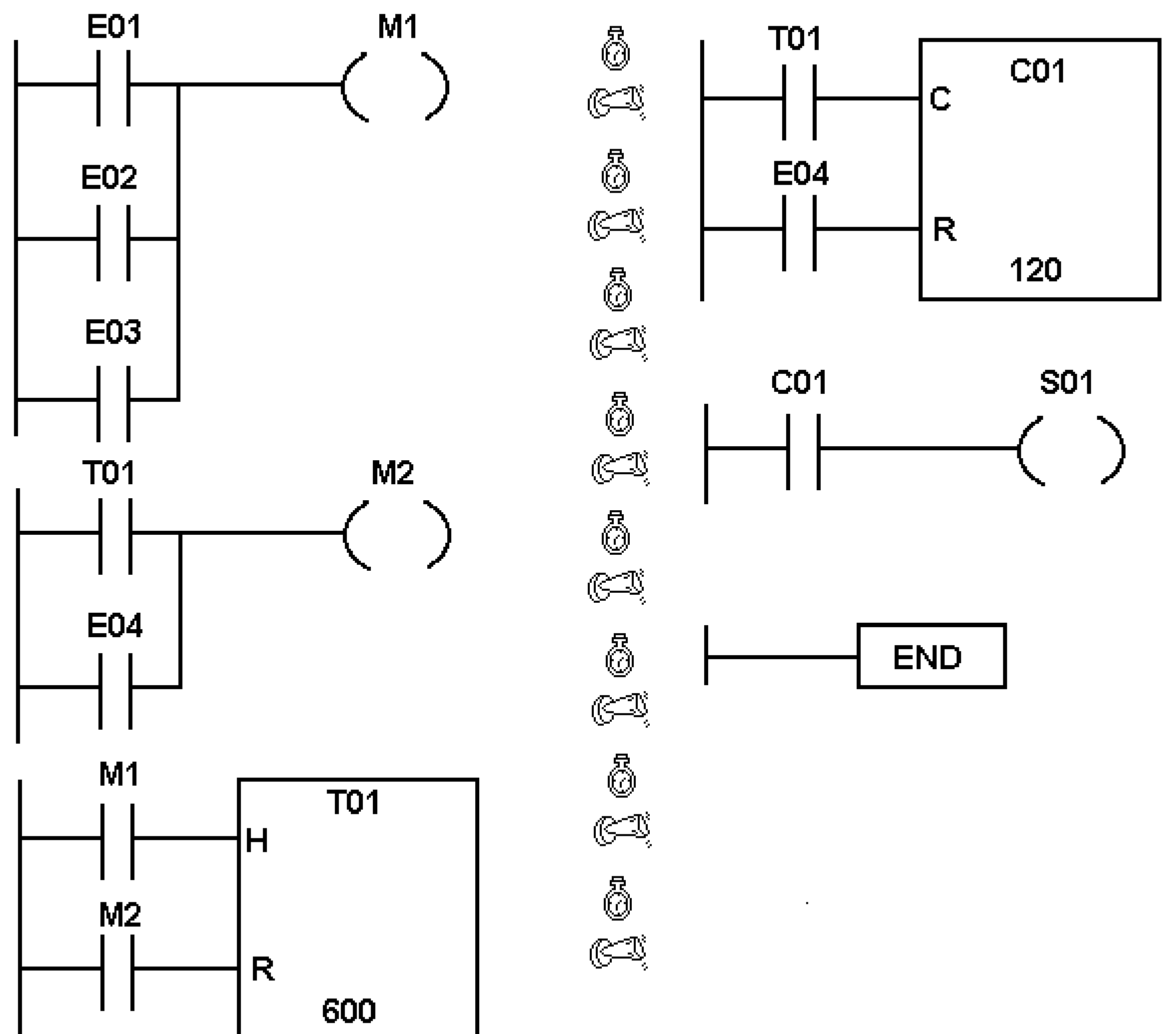


Figura 4-98 Solução ladder tempo de falha

4.6 Funções SET e RESET

As funções de saída SET e RESET controlam os bits de endereçamento de forma independente. A função SET energiza o bit de memória/saída endereçado, e a função RESET desenergiza o bit endereçado. Estas são funções de retenção, ou seja, estes elementos assumem seu valor sempre que sua entrada estiver energizada, mantendo seu valor até que sua função complementar seja acionada.

A representação desta função é a seguinte:



Figura 4-99 Representação funções SET e RESET

O funcionamento das funções SET e RESET são detalhados através dos gráficos da figura 4-32.

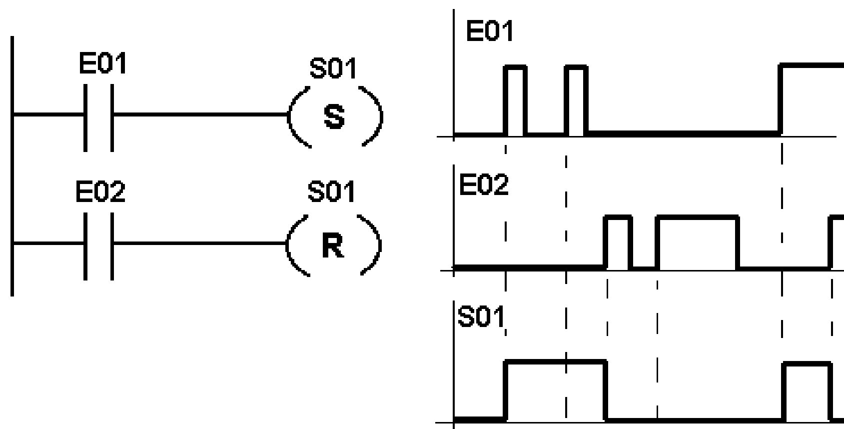


Figura 4-100 Funcionamento de SET e RESET

Quando um relê bobina e as funções set e reset são utilizadas em conjunto para um mesmo endereçamento, o valor assumido pelo bit endereçado corresponde ao último comando assumido por um destes elementos dentro do mesmo ciclo do programa. Na figura 4-33 o valor da função set e reset serão ignorados, pois a linha três do programa utiliza um comando bobina, a atualização do bit ocorre conforme o estado desta lógica de programa.

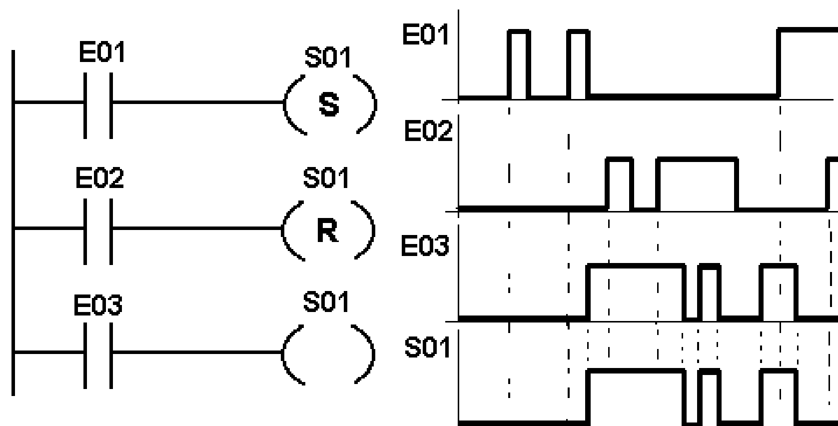


Figura 4-101 Comparação set, reset e bobina

4.7 Exercícios

4.7.1 Exercício 1: Semáforo

Definir um programa em ladder para um semáforo, onde temos os seguintes sinais:

- E01 : Ha: habilita circuito,
- S01 : VM1: Lâmpada vermelha 01,
- S02 : AM1: Lâmpada amarela 01,
- S03 : VD1: Lâmpada verde 01,
- S04 : VM2: Lâmpada vermelha 02,
- S05 : AM2: Lâmpada amarela 02,
- S06 : VD2: Lâmpada verde 02.

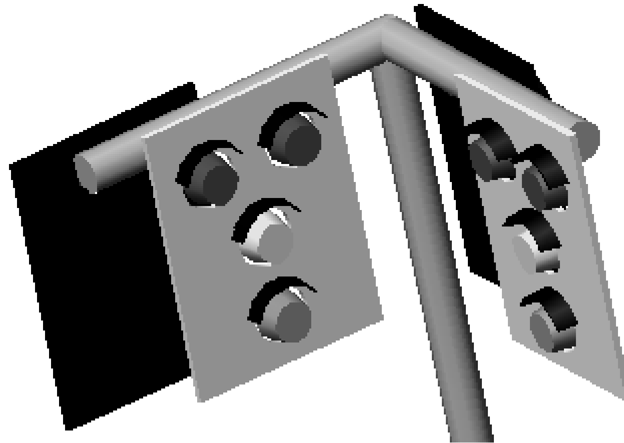


Figura 4-102 Semáforo

O algoritmo de controle é o seguinte:

- Ligar a lâmpada verde durante 10 segundos;
- Ligar a lâmpada vermelha durante 13 segundos;
- Ligar a lâmpada amarela durante 3 segundos;
- Utilizar a seguinte seqüência: primeiro ligar a lâmpada verde, após tempo de verde, ligar a lâmpada amarela, após tempo de amarela ligar a lâmpada vermelha. Ao ligar lâmpada vermelha também ligar a lâmpada verde do outro semáforo. Veja diagrama da figura 4-35.

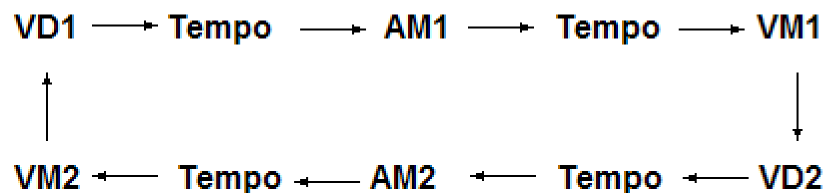


Figura 4-103 Diagrama de tempo do semáforo

4.7.2 Exercício 2: Máquina seqüencial

Definir um programa em ladder para a máquina sequencial de envase de líquidos, representada na figura 4-36. Onde temos os seguintes sinais:

- E01 : Ha: habilita máquina,
- E02 : S1: sensor ótico de presença de recipiente na entrada,

- ➲ E03 : S2: Sensor ótico de presença de recipiente no bico,
- ➲ E04 : S3: Sensor ótico de presença de recipiente na saída,
- ➲ S01 : M1: contator aciona motor 1 da correia de transporte,
- ➲ S02 : Bico: Aciona bico de dosagem.

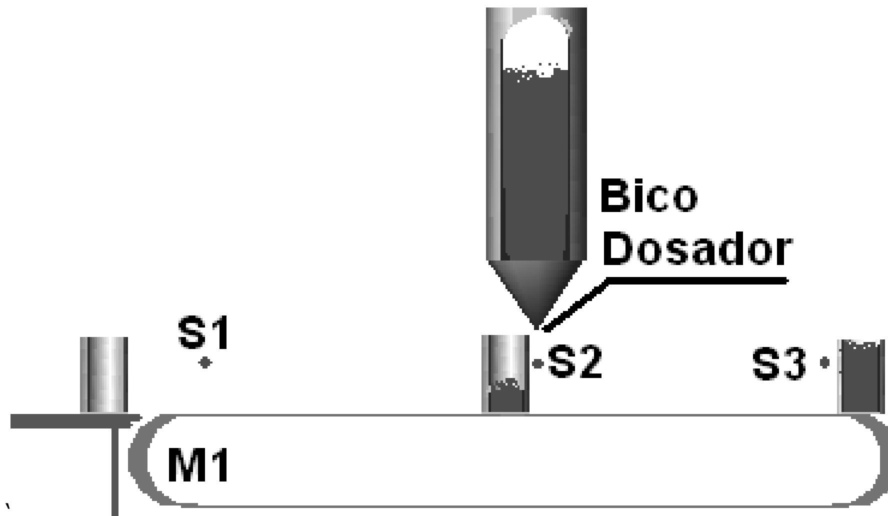


Figura 4-104 Máquina sequencial de envase

Utilizar o seguinte algoritmo de controle:

- Ao habilitar a máquina seguir a seqüência abaixo;
- Ligar motor para posicionar produto no bico, caso sensor de saída livre e produto na entrada;
- Quando existir produto no bico (S2=1) acionar válvula durante 3 segundos;
- Ao encher a lata, desligar o bico e esperar 1 segundo;
- Após espera, ligar o motor até atingir o sensor de saída;
- Ligar motor para posicionar novamente o produto no bico, caso sensor de saída livre e produto na entrada;
- Repetir passos anteriores, enquanto existir recipiente na entrada e a máquina estiver habilitada.

4.7.3 Exercício 3: Piscar Lâmpada de Alarme

Para o exemplo 4.5.4 modificar o controle do acionamento da lâmpada de alarme para chamar mais atenção, deixando a lâmpada piscando enquanto existir alarme. Sendo o tempo de lâmpada ligada de 2 segundos e desligada em 1 segundo.

Dica para resolução do exercício: criar um circuito (lógica) independente com dois temporizadores, para controle de um bit auxiliar de memória.

5 - Máquina de Estado

"Somente Hoje

dedicarei dez minutos de meu tempo a uma boa leitura, lembrando-me de que, assim como alimento é necessário para o corpo, a boa leitura é necessária para a alma."

João XXIII apud Puebla, 1997, p.100

Neste capítulo analisaremos os conceitos envolvidos na utilização de máquinas de estado, abordando os conceitos de utilização e como transportá-lo para a linguagem ladder. A máquina de estado também é conhecida por diagrama de estado.

5.1 Definição

Máquina de estado é uma ferramenta gráfica de auxilio a programação que permite representar o funcionamento de um sistema através de estados de funcionamento e as transições entre os estados.

5.1.1 Estado

O estado é o espaço reservado do diagrama que define uma ação no acionamento do sistema. A todo estado é associada no mínimo uma ação e um número que identifica este estado.

O número associado ao estado deve ser único, não podendo ser repetido a outro estado dentro do mesmo diagrama. A numeração dos estados não precisa ser seqüencial e deve ser utilizada a melhor forma de organização

que o programador encontrar, utilizando sempre os números inteiros positivos a partir do número 1 (um).

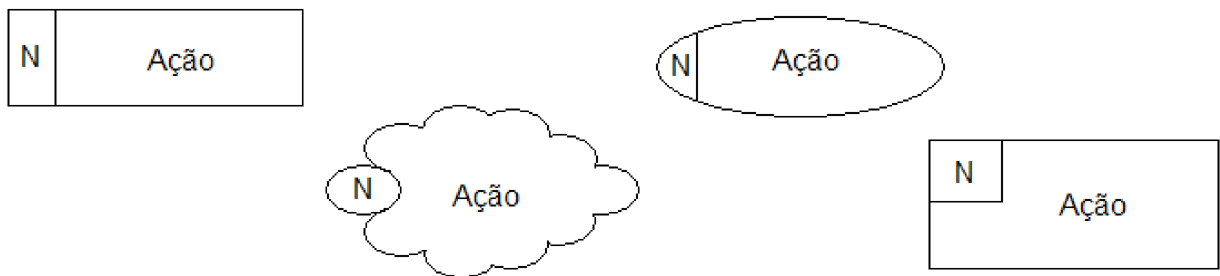


Figura 5-105 Representação do estado

O estado é representado por uma figura geométrica fechada dividida em duas partes, uma parte menor com o número do estado e uma parte maior para a descrição da ação referente ao estado. A figura 5-1 mostra algumas formas para representar o estado. Neste trabalho adotamos a representação do estado através de um retângulo.

5.1.2 Transição

A transição é a ligação entre dois estados. A mudança de um estado ativo para outro somente ocorre quando a condição associada à transição for verdadeira.

Podemos associar varias condições para uma mesma transição utilizando as operações lógicas "e" (AND) e "ou" (OR).

A transição é representada por uma seta que liga os dois estados, identificando a condição necessária para a validação da transição. A figura 5-2 demonstra como representar a transição entre dois estados.

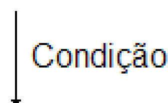


Figura 5-106 Representação da transição

Para simplificar o diagrama, deixando-o mais limpo, utilizamos algumas definições para a representação do diagrama de estado:

- A comparação com o número um é omitida. Ex: se condição for "x=1", representamos apenas "x"

- Caso a operação lógica utilizada entre as condições for à operação “e”, também omitimos esta operação. A figura 5-3 mostra esta representação.

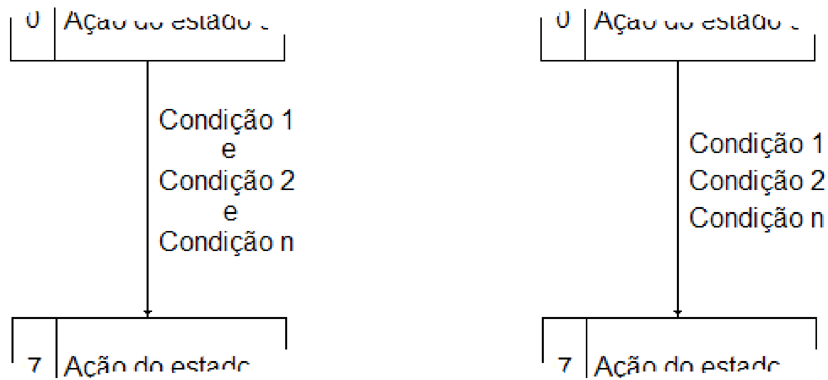


Figura 5-107 Simplificação lógica E na transição

- A operação lógica “ou” é simplificada utilizando-se de outra transição entre os dois estados. A figura 5-4 demonstra como fazer esta simplificação.

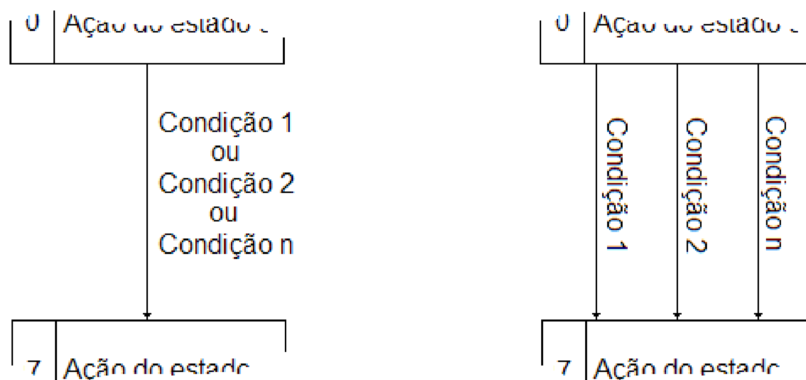


Figura 5-108 Simplificação lógica OU na transição

- Um estado pode ter transições diferentes para ligar outros estados. Na figura 5-5 podemos ver uma associação de diversos estados.

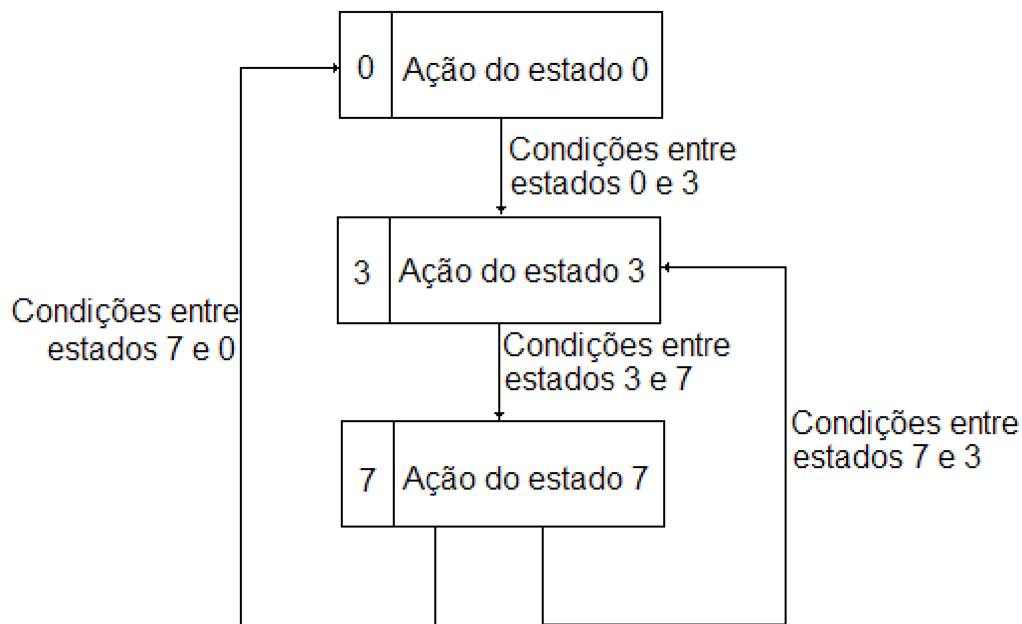


Figura 5-109 Exemplo diagrama de estado

5.1.3 Estado Inicial

O estado inicial corresponde ao primeiro estado do sistema de onde partem todas as analises de funcionamento deste. O estado inicial é diferenciado através de sua numeração. O número do estado inicial é o zero.

O sistema automático que executa o diagrama de estado deve ser capaz de inicializar o sistema sempre com o estado inicial.

5.2 Exemplos de Diagrama de Estado

Para facilitar a aplicação dos conceitos sobre diagrama de estado apresentamos alguns exemplos e técnicas especiais para elaboração destes diagramas.

5.2.1 Exemplo: Chave de Partida Direta

O diagrama de estado de uma chave de partida direta é representado na figura 5-6, onde no estado zero que corresponde à condição inicial o motor esta desligado. No estado um o motor ficará ligado. A transição do estado zero para

o um somente ocorre quando o botão de emergência e botão liga forem liberados, ou seja, estiverem com o valor um.

Retorna para o estado zero quando o botão de emergência ou o botão desliga estão com valor zero, retornando ao estado Zero e conseqüentemente desligando o motor.

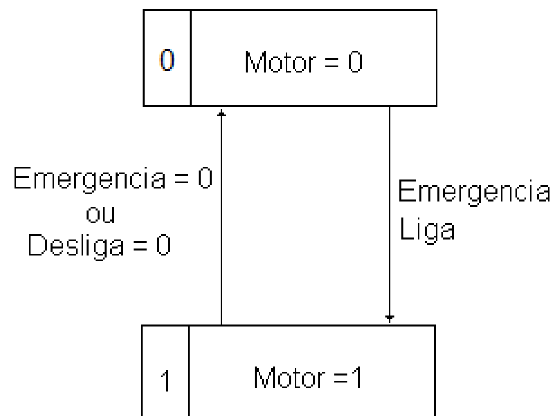


Figura 5-110 Diagrama de estado: chave partida direta

5.2.2 Exemplo: Controle de Nível com Forçamento de Estado

O diagrama de estado de um sistema de controle de nível de um tanque é representado pela figura 5-7, onde observamos que no estado zero, condição inicial, a bomba permanece desligada. No estado um o motor ficará ligado. Nos estados dois e três a bomba permanece inalterada, enquanto o sistema fica aguardando um atraso de tempo para avançar para o próximo estado. Este atraso de tempo é necessário para evitar a queima do motor elétrico da bomba, pois enquanto o sensor ficar com o nível oscilando em torno de si a bomba permanece com o seu acionamento anterior evitando o chaveamento constante desta.

A transição do estado zero para o dois se dá de forma imediata com a identificação do sinal de nível baixo e chave Habilita igual a um, inicializando-se a contagem do tempo de atraso, caso o sinal de nível retorne a indicar presença de líquido antes do tempo de atraso o sistema retorna para o estado zero, caso contrario ao finalizar o tempo o sistema avança para o estado um, ligando a bomba.

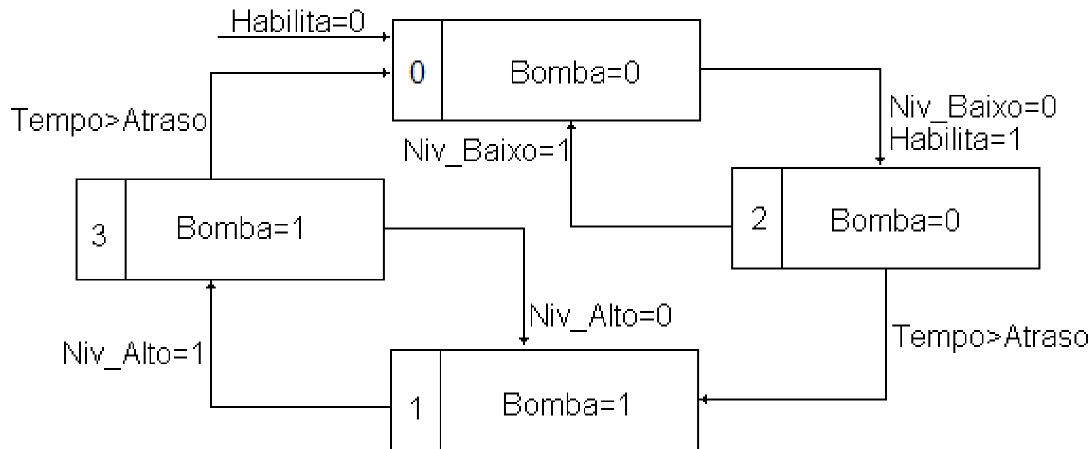


Figura 5-111 Diagrama de estado: controle de nível

Após acionada a bomba no estado um, o sistema desliga quando identifica a presença de nível cheio, quando migra para o estado três, esperando o tempo de atraso para se posicionar no estado zero. Se a presença do sinal for menor que o tempo de atraso, o sistema retorna para o estado um, mantendo a bomba ligada.

No estado zero temos uma transição em aberto, indicando que a qualquer momento se a chave habilita for igual a zero o sistema força o retorno imediato ao estado zero.

5.2.3 Exemplo: Máquina Sequencial com Múltiplos Diagramas

Para o exercício proposto no item 4.7.2 (Máquina Sequencial de dosagem), vimos que o sistema é composto por dois atuadores, ou seja, correia de transporte e bico dosador. Para facilitar a elaboração dos diagramas de estados deve-se criar um diagrama individual para cada grupo de atuador, chamando cada diagrama de máquina. Neste caso teremos duas máquinas, a máquina número 1 correia de transporte e a máquina número 2 bico dosador.

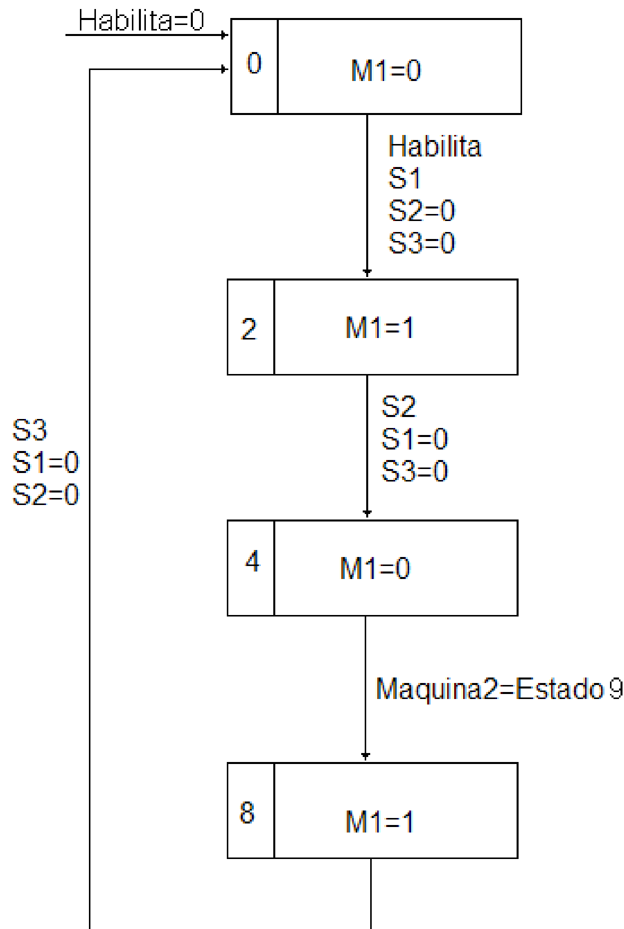


Figura 5-112 Diagrama de estado: sistema de dosagem - transporte

O diagrama de estado da máquina número um está na figura 5-8. Neste caso o controle do acionamento do motor foi feito com quatro estados. No estado zero o motor é desligado. A transição do estado zero para o estado dois ocorre quando a máquina estiver habilitada e o sensor de entrada estiver igual a um, os outros sensores (S2 e S3) devem estar em zero. No estado dois o motor fica energizado. A migração para o estado quatro ocorre quando o sensor do bico for igual a um e os outros sensores iguais a zero, neste estado o motor permanece desligado. A migração para o estado oito ocorrera quando o outro diagrama (máquina 2) estiver no estado nove, neste estado o motor será ligado novamente. O sistema retorna para o estado zero quando o sensor de saída for igual a um e os outros sensores forem iguais a zero. O sistema também retorna para o estado zero quando a chave habilita for desligada, forçando a entrada no estado zero.

O diagrama de estado da máquina número dois é visualizado na figura 5-9. Esta máquina também é composta por quatro estados. No estado o zero o bico fica desligado e será ligado somente no estado dois. A transição do estado zero para o dois ocorre caso a máquina esteja habilita, o sensor do bico igual a um e o diagrama da correia (máquina 1) esteja no estado quatro. No estado dois abre o bico de dosagem e espera três segundos antes da transição para o estado quatro onde o bico de dosagem fica desligado. Neste estado (quatro) será temporizado mais 1 segundo, que seria um tempo de acomodação do produto na lata, para migrar ao estado nove. A migração do estado nove para o zero ocorrerá quando a lata sair da frente do sensor S2. Como segurança a máquina também será desligada caso o chave desabilite o sistema (Habilita=0), ou ocorra uma falha no sensor S2.

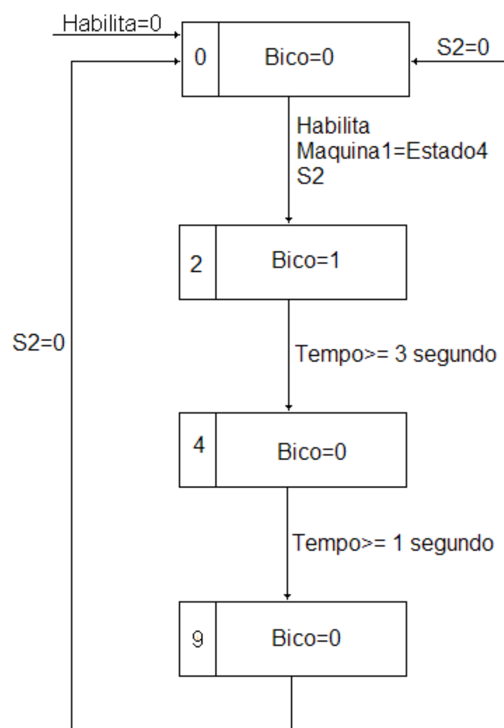


Figura 5-113 Diagrama de estado: sistema de dosagem – bico dosador

5.3 Utilizando Máquina de Estado com CP

A utilização de diagramas de estados na programação de CP facilita a manutenção do sistema automático, pois permite a procura de falhas mais rapidamente, visto que o número do estado da máquina pode ser apresentado numa IHM.

A utilização de máquina de estados para CP ocorre de duas maneiras, a primeira mais fácil, mas nem sempre disponível, é converter o diagrama de estado diretamente para a linguagem SFC, caso o sistema de programação do CP permita a utilização desta linguagem. Caso não existe esta possibilidade o diagrama de estado deverá ser convertido para a linguagem ladder.

5.3.1 Convertendo Diagrama de Estado em Ladder

Para converter o diagrama de estado em linguagem ladder deve-se:

- ➲ Garantir que o estado inicial seja inicializado com a energização do CP.
- ➲ Reservar um *flag* (operando binário) para cada estado, ficando ativo enquanto o estado estiver ativo. Caso o CP não possua flags suficientes para representar o seu sistema, para simplificar a utilização de memórias, pode-se utilizar apenas a função de comparação. Neste caso o ciclo de programa ficará maior.
- ➲ Reservar um registrador (operando de 16 bits) para indicar o número do estado ativo, este operando facilita a manutenção do sistema, pois será apresentado via IHM.
- ➲ Ao efetuar uma transição incluir o *flag* que identifica o estado atual na comparação.
- ➲ Acionar os dispositivos, referentes à ação do estado, com base no *flag* do estado.

Portanto a conversão de um diagrama de estado começa pela verificação da condição de inicialização do CP, onde deverá ser movimentado o valor zero (estado zero) para todos os registradores de indicação do estado. Neste momento, inicializa-se, zeram-se todos os flags de indicação de estado, setando-se apenas o *flag* que indica o estado zero.

Na execução de um estado deve-se verificar se o *flag* do estado está ativo, ou comparar se o valor do registrador é igual ao número do estado, para utilizar as funções de SET e RESET ativando as ações deste estado.

Na transição de um estado a outro se verifica o estado atual, através da comparação do registrador ou do *flag* indicativo do estado, verificar as condições da transição. Caso estes itens estejam ativos deve-se mover o número do novo estado para o registrador e trocar o *flag* ativo do estado, ou seja, resetar o *flag* do estado antigo e setar o *flag* do estado novo.

ATENÇÃO: Nos exercícios deste livro reservaremos os registradores D11 a D19 para referenciar o estado das máquinas (diagramas de estados) 1 a 9. E reservaremos os flags M100 a M199 para indicar o estado ativo, verificado que M100 a M109 corresponde aos estados 0 a 9 da máquina 1.

A figura 5-10 indica o programa em ladder criado a partir de um diagrama de estado. Nesta figura são apresentadas três colunas, na primeira mostramos o diagrama de estado e na segunda e terceira como fica o ladder a partir deste diagrama, note que são duas soluções diferentes, uma que utiliza apenas flags e outro que utiliza comparadores. Para enfatizar as diferenças esta conversão foi dividida em quatro linhas. Na primeira linha foram colocadas as lógicas de inicialização do estado no ladder, onde na primeira linha não temos um correspondente no diagrama de estado, pois esta definição já é implícita ao diagrama que inicia sempre no estado zero. Na segunda linha temos a forma do estado zero, ou seja, neste estado desligamos o motor e indicamos que estamos no estado zero. Na terceira linha visualizamos como fazer as comparações e verificações das condições para migrar para outro estado.

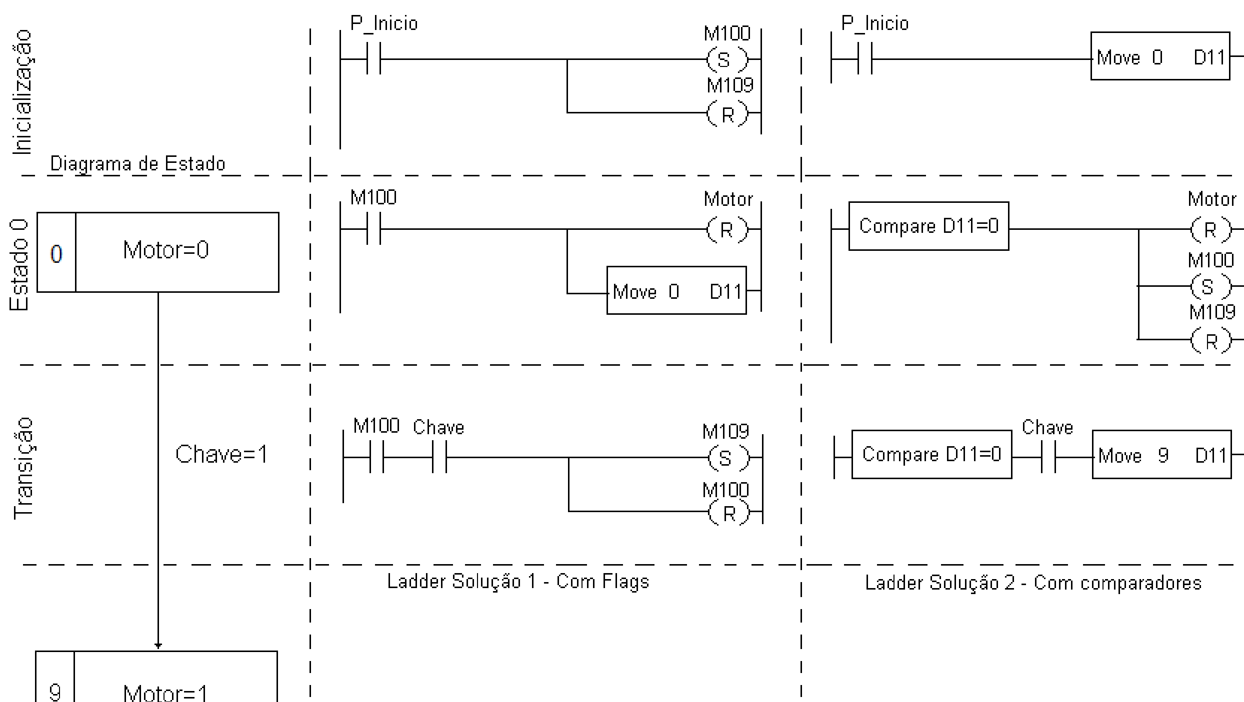


Figura 5-114 Ladder a partir do diagrama

5.3.2 Exemplo: Ladder do Diagrama da Máquina Seqüencial

Este exemplo serve para melhorar o entendimento da conversão de diagrama de estado para ladder. Vamos converter os diagramas de estado da máquina seqüencial das figuras 5-8 e 5-9, resolvido no exemplo 5.2.3.

A solução é dada nas figuras 5-11 e 5-12, onde “P_Inicio” corresponde a um *flag* do sistema que indica que o CP foi energizado pela primeira vez, este *flag* fica acionado apenas durante o primeiro ciclo do programa.

Quanto aos dois programas ladder obtidos a partir do diagrama de estado, observar que eles seguem uma lógica de controle seqüencial de tal forma:

- Primeira Lógica: Lógica de inicialização e entrada no estado zero.
- Segunda Lógica: Executa a ação referente ao estado zero.
- Terceira Lógica: Realiza a transição entre o estado zero com o dois.
- Quarta Lógica: Realiza as ações referentes ao estado dois.
- Quinta Lógica: Realiza a transição entre o estado dois com o quatro.
- Sexta Lógica: Realiza as ações referentes ao estado quatro.
- Sétima Lógica: Realiza a transição entre o estado quatro com o seguinte.
- Oitava Lógica: Realiza as ações referentes ao último estado.

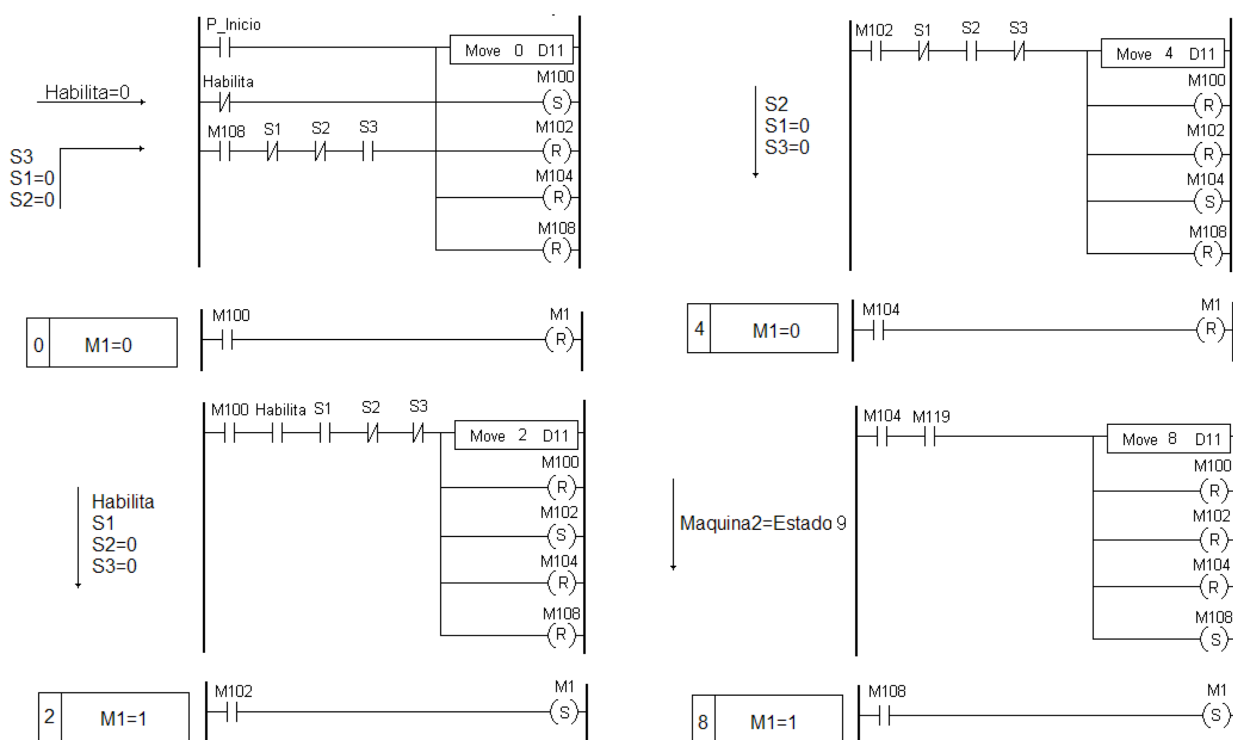


Figura 5-115 Ladder correia da máquina seqüencial

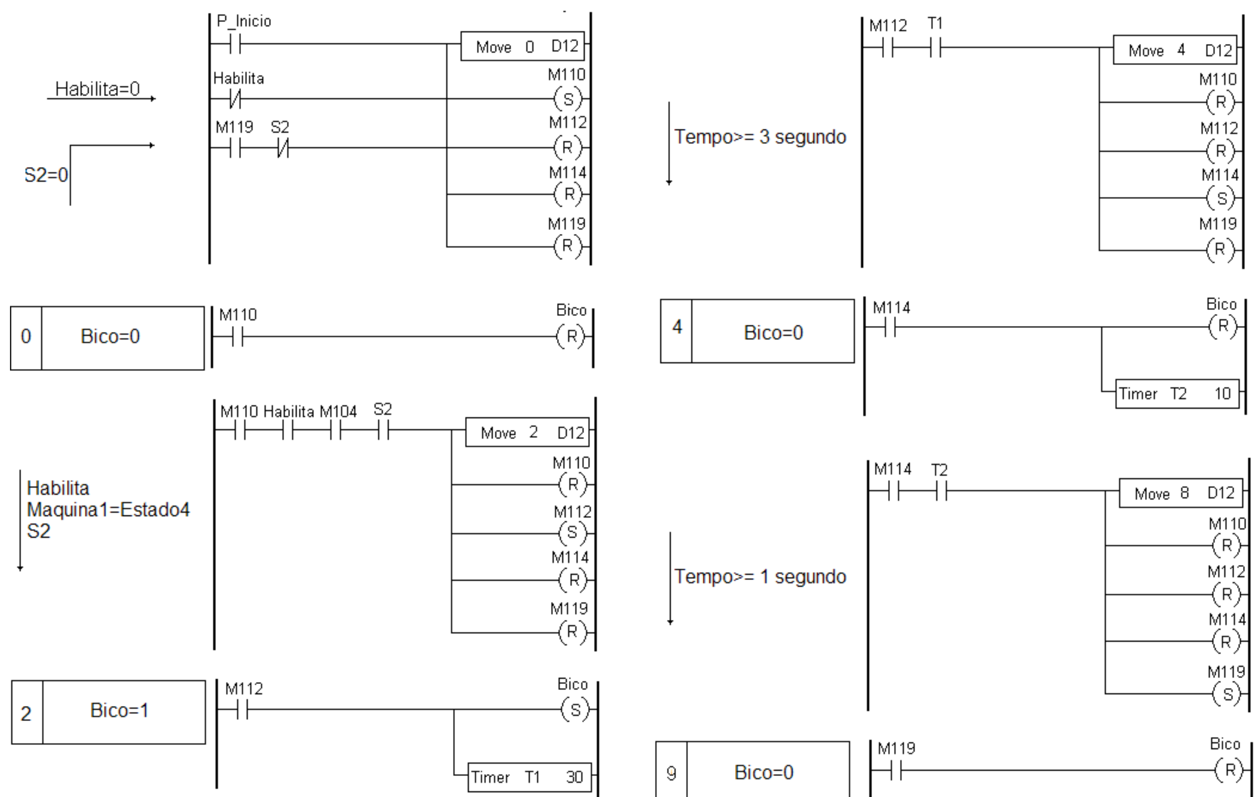


Figura 5-116 Ladder bico dosador da máquina seqüencial

5.4 Exercícios

5.4.1 Exercício 1: Desbobinador e Corte Automático

Definir diagramas de estados e programa ladder para controle de uma máquina desbobinadora com guilhotina para corte automático, onde temos os seguintes sinais:

- E01 : B1: Botão liga sistema,
- E02 : B0: Botão desliga sistema,
- E03 : Pulso: Entrada de contagem de pulsos,
- S01 : M1: Motor desbobinador,
- S02 : V1: Válvula de acionamento da guilhotina.

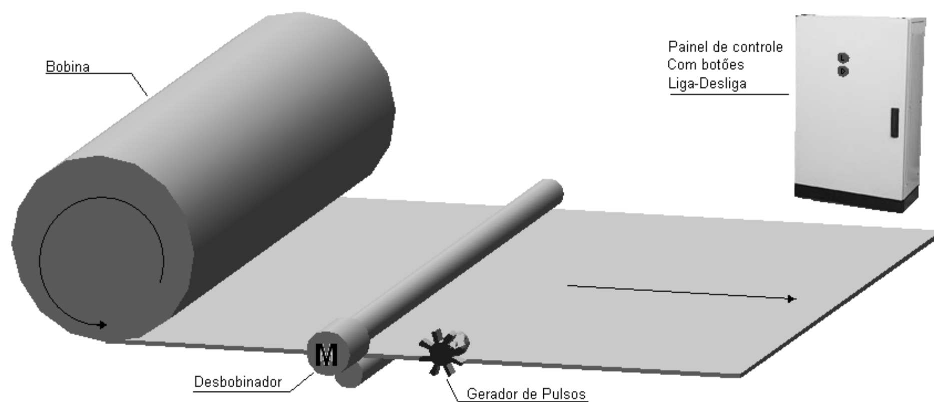


Figura 5-117 Desbobinador e corte automático

Procedimentos para controle:

- O sistema será habilitado sempre que for dado um pulso em B1;
- O sistema deverá ser desabilitado sempre que for dado um pulso em B0;
- Quando o sistema estiver habilitado ele deve ligar o motor;
- Após 10 pulsos no gerador de pulsos deve-se desligar o motor e acionar a guilhotina;
- A guilhotina deve ficar acionada por 1 segundo;
- Após finalização do tempo de guilhotina o painel deverá ser desabilitado automaticamente.

6 - Manutenção

"Sabe que não acredito que alguém jamais tenha ensinado alguma coisa a outro. Duvido da eficácia do ensino. A única coisa que sei é que quem quiser aprender, aprenderá. E talvez o professor seja um fator que facilite, uma pessoa que apresenta as coisas e mostra aos outros como é empolgante e maravilhoso, e os convida a provarem."

ROGERS apud BUSCAGLIA, 1996, p.23

A automação é um processo de engenharia que envolve o desenvolvimento de mecanismos com acionamentos pré-definidos através de um programa para controladores lógicos. Este programa é desenvolvido via computador pessoal, assim como a maioria dos dispositivos eletrônicos também possuem configuradores via computador. Um sistema de automação é completado por outras informações além dos programas, também fazem parte da automação os desenhos técnicos, as especificações e as literaturas técnicas. E para agilizar um processo de manutenção é importante que todas estas informações estejam acessíveis de forma organizada e segura.

Este capítulo tem como objetivo orientar os profissionais responsáveis por manutenção e projetos na importância da organização de programas e configurações em um micro computador, normalmente um laptop, bem como organizar toda a documentação impressa. No final do capítulo abordaremos a temática referente à identificação de falhas.

6.1 Organização de Arquivos

Como a maioria dos dispositivos eletrônicos possui configuração via software, faz-se necessário à organização destes dados para a redução do tempo de manutenção de uma máquina/processo. Deste modo devem-se organizar os softwares executáveis e os aplicativos em um microcomputador, bem como as configurações de todas as máquinas, tais como: conversores, inversores, controladores de temperatura e servos-conversores.

A organização destes dados deve seguir a um padrão de armazenamento de dados, para facilitar a recuperação dos dados em disco. Para atingir este objetivo de organização existem vários modos. Para quem nunca pensou nesta possibilidade sugiro uma maneira de organizar os dados. Este método consiste em criar diretórios individuais e separados para dados e programas executáveis que são instalados. No primeiro caso criamos um diretório chamado DADOS e no segundo caso um ORIGINAL (Figura 8-1). O diretório DADOS deverá ser utilizado para armazenar parâmetros e programas do sistema, e para melhorar a organização destes dados, este diretório deve ser dividido em três partes: ELÉTRICA, PARTICULAR e ADMINISTRAÇÃO. Por sua vez estes diretórios subdivididos conforme a necessidade do ambiente de trabalho.

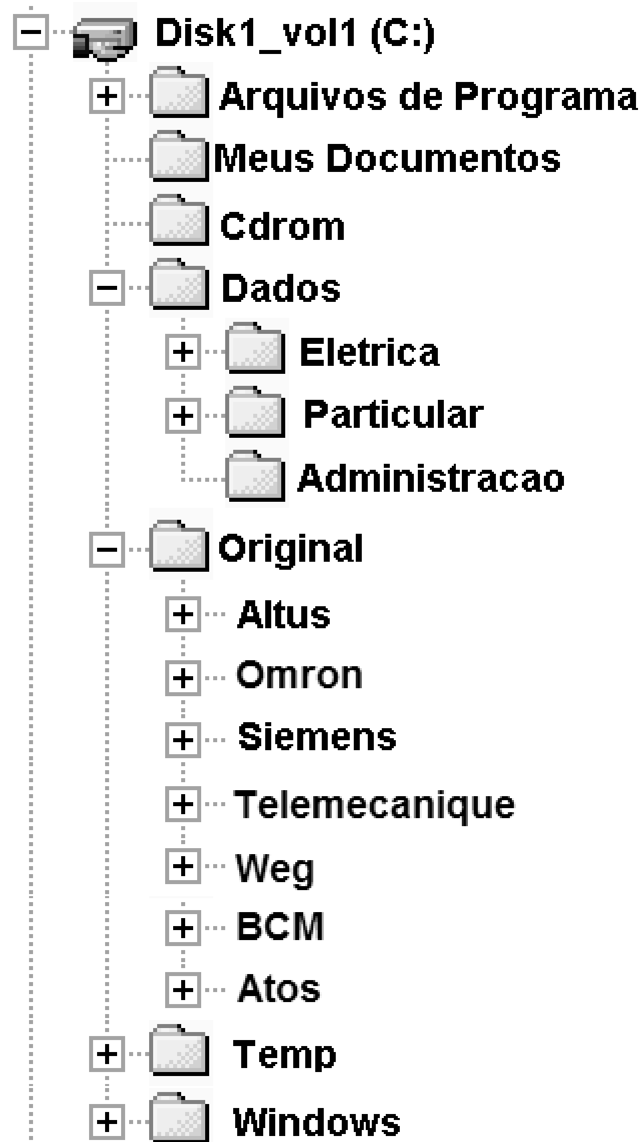


Figura 8-118 Estrutura de arquivos - inicio raiz

O diretório DADOS\ELETRICA (Figura 8-2) é subdividido em vários subdiretórios, um para cada aplicativo e ou configuração. O nome de cada diretório corresponde ao nome da máquina/processo. Como exemplo observe o caso da PRENSA XYZ, que utiliza controlador lógico e inversores de frequências, além de programa de supervisão. Neste caso foi criada a pasta PRENSA XYZ com os subdiretórios LADDER, SUPERVISORIO,

INVERSORES e MANUAIS. No diretório LADDER armazena-se o programa em ladder do aplicativo, no SUPERVISORIO o arquivo de configuração da aplicação bem como todos os arquivos fontes utilizado. O diretório INVERSORES é utilizado para armazenar as configurações dos inversores. Por último o diretório MANUAIS é utilizado para armazenar todos os manuais de funcionamento da máquina fornecidos pelo fabricante em meio digital.

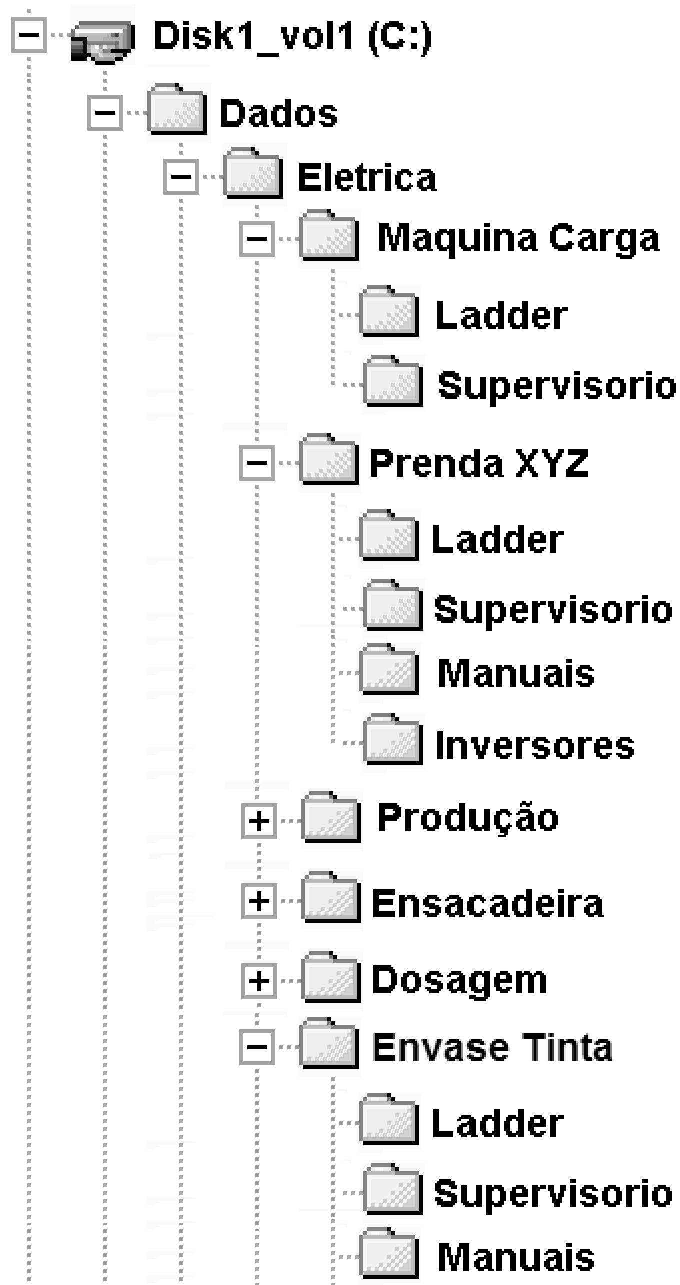


Figura 8-119 Estrutura de arquivos - inicio elétrica

No diretório ORIGINAL (Figura 8-1) armazenam-se todos os dados fornecidos pelo fabricante dos softwares que são utilizados. Para a organização entre eles, usam-se subdiretórios com os nomes de cada fabricante, que serão novamente subdivididos através de subdiretórios representando cada aplicativo

e manuais, como segue na figura 8-3. Na pasta aplicativo armazena-se o software fonte instalável, caso este seja fornecido de forma informal e dispensando licença de uso (freeware e shareware). A subpasta manuais é utilizada para guardar a documentação do software, ou seja, manuais e tutoriais.

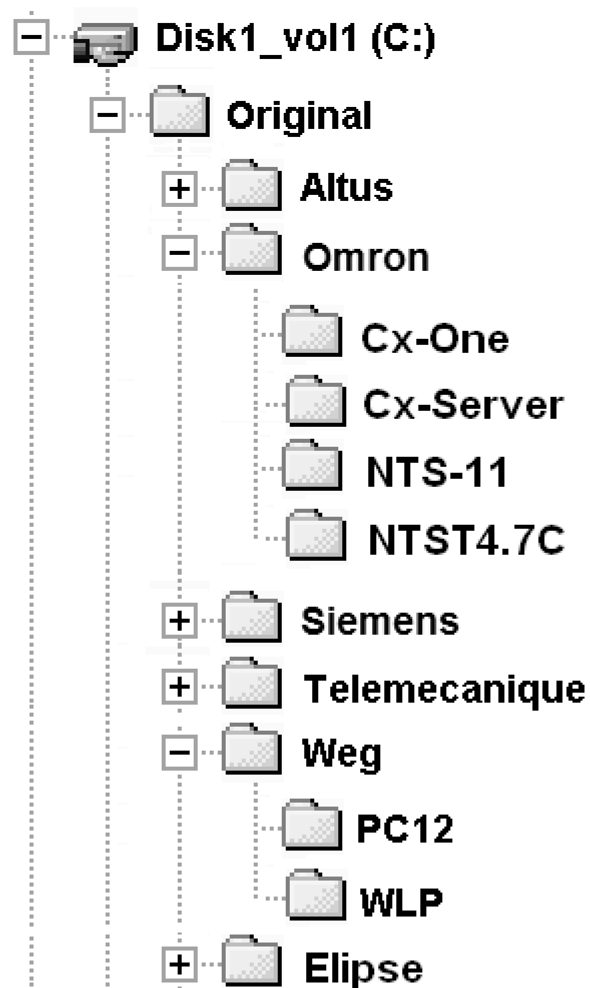


Figura 8-120 Estrutura de arquivos - inicio original

6.2 Segurança dos Dados

Além da organização dos dados devem-se tomar alguns cuidados para preservar a integridade do armazenamento destas informações, sem riscos de erros de leituras e perdas dos dados e/ou mídia de armazenamento. Para a segurança no armazenamento das informações devem-se tomar alguns cuidados:

- Armazená-los em mais de uma mídia.
- Guardá-los em locais físicos distantes um do outro, por exemplo uma cópia ficar no setor de manutenção e outra no prédio de administração. Para profissionais liberais uma cópia em casa a outra no escritório.
- Verificar a possibilidade de utilizar CDROM/DVD de qualidade como segunda mídia, pois além de ser um meio bastante confiável, também é um sistema de preço acessível, rápido e seguro.
- O Disco rígido do computador não é totalmente seguro, nunca utilize apenas o seu próprio computador como backup do sistema, caso você não disponha de gravador de CDROM/DVD, faça o backup em *pendrive* e no disco rígido de outro computador.
- Para transporte de dados entre micros e/ou para viagem pode-se utilizar os *driver* com flash-eprom (PENDRIVE), e também a caixa do correio eletrônico e/ou disco virtual (servidor internet) para transportá-lo.

6.3 Organizar Manuais e Apostilas

A utilização de literaturas técnicas para a manutenção de máquinas e processos é fundamental e deste modo é fundamental a organização de todos os manuais e apostilas dos equipamentos em um espaço de fácil acesso, como por exemplo um armário com pastas arquivos. Seguindo o exemplo dos aplicativos estes manuais e apostilas são separadas em pastas por fabricantes e ou máquinas. O mais importante é tê-los de forma rápida e que não se utilize muito tempo para achá-los.

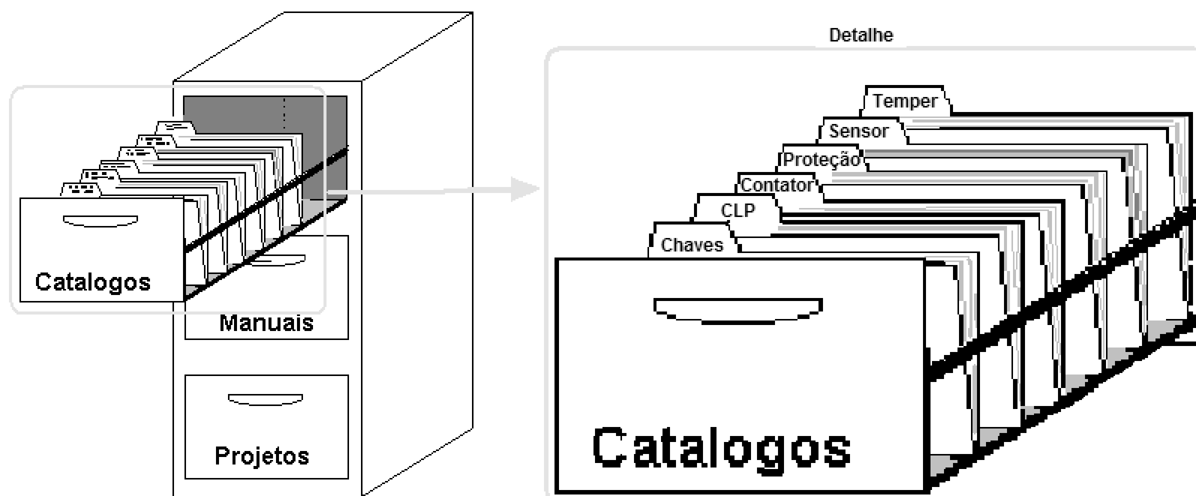


Figura 8-121 Arquivos para projetos,catálogos e manuais

Se estiver faltando algum material, entre em contato com o seu fornecedor e solicite os materiais que estão faltando.

6.4 Documentação dos Aplicativos

É importante também ter em mãos a documentação referente à máquina/processo, tais como manual de operação e esquema elétrico, guardando-os em local de fácil acesso. Não se esquecer de solicitá-los quando estiver realizando um processo de compra de uma máquina e ou processo.

6.5 Como Identificar Falhas Elétricas

O responsável pela manutenção elétrica de um sistema automático deve seguir algumas regras para identificar falhas elétricas de uma forma mais eficiente e segura. Algumas destas regras serão observadas mesmo antes de ocorrer à falha elétrica, tais como:

- ➲ Conhecer o esquema elétrico da máquina, de tal forma que saiba, previamente, qual foi o padrão utilizado para a numeração de fios e nomes de dispositivos.
- ➲ Conhecer o manual de operação da máquina, identificando as particularidades de funcionamento da máquina/processo.
- ➲ Conhecer o manual de programação e hardware do equipamento, a fim de comunicar e identificar falhas do equipamento. É importante

saber como se comportam os sinalizadores visuais (led) de funcionamento.

- ➲ Ver como o operador comanda a máquina, dedicar um período para o funcionamento da máquina e confrontar o manual de operação com a forma utilizada para operação.
- ➲ Manter o painel elétrico organizado, ao finalizar a manutenção deixar o painel limpo, colocando todos os fios dentro das calhas, numerar os fios e atualizar o esquema elétrico.

Após ocorrer o problema e observado as regras anteriores, complementar com os seguintes itens:

- ➲ Verificar relato do problema;
- ➲ Simular o problema se possível;
- ➲ Verificar as tensões elétricas;
- ➲ Verificar botoeiras de comando;
- ➲ Verificar estado do controlador e equipamentos (ver sinalizadores visuais do controlador);
- ➲ Monitorar o software;
- ➲ Substituir dispositivos defeituosos.

O primeiro ponto para uma manutenção rápida é identificar corretamente o problema. O segundo passo é identificar as possíveis causas, isolando-as uma a uma até resolver a(s) falha(s).

Não devemos esquecer que utilizar o suporte ao cliente dos fornecedores, através de telefone, internet e visita presencial de um técnico da empresa, neste caso é importante manter o cadastro de fornecedores sempre atualizado.

6.6 Melhorias em Máquinas

Normalmente as máquinas antigas além de não possuir uma produção adequada aos padrões atuais, possuem um painel elétrico ineficiente, que demanda com freqüência um tempo enorme de manutenção. Para diminuir o tempo de máquina parada é aconselhável modernizar a instalação elétrica, substituindo os atuadores desgastados com o tempo e melhorando o aspecto visual interno do painel. Neste processo devem-se utilizar tecnologias novas

diminuindo o número de componente e aumentando a vida útil dos componentes elétricos.

A figura 8-5 representa uma foto de um painel elétrico antigo, onde observamos as seguintes deficiências:

- Painel é pequeno;
- Falta de canaletas, o que dá um aspecto feio a disposição dos condutores;
- Falta de identificação dos condutores;
- Falta de conectores tipo borne;
- Falta de documentação (Projeto elétrico).

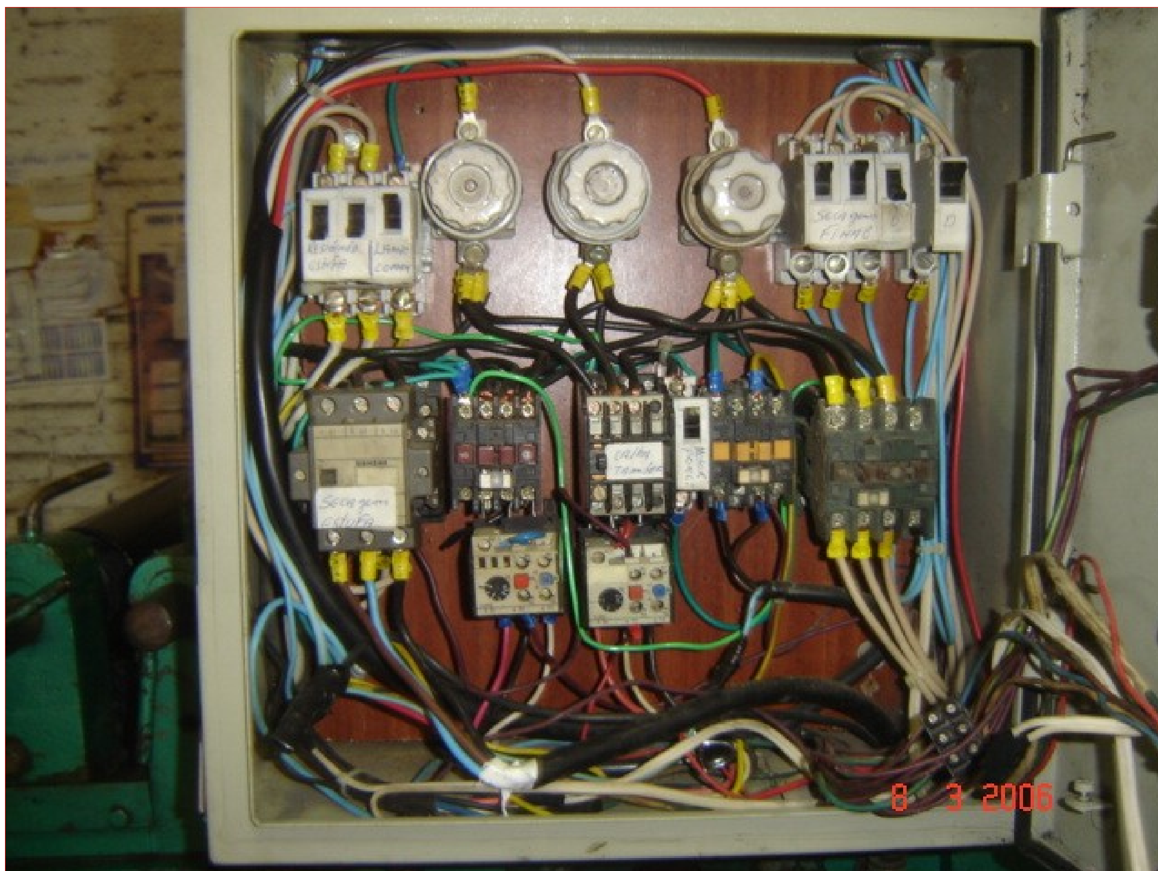


Figura 8-122 Painel elétrico antigo

As deficiências apontadas na figura 8-5 podem ser corrigidas, utilizadas novas tecnologias, tais como:

- Atualização do esquema elétrico;
- Marcadores de fios (anilhas);
- Bornes (conectores);
- Disjuntor motor, para reduzir espaço ocupado;
- Canaletas, para esconder o condutor e dar um aspecto visual melhor ao painel;
- Controlador lógico que pode permitir maior flexibilidade na utilização do mesmo.

Efetuando estas correções no painel a funcionalidade e o aspecto visual do painel será melhor, o que evitará paradas constantes para manutenção, devido aos novos equipamentos. E quando a máquina parar para manutenção, o tempo de parada será mais curto, pois teremos a documentação atualizada e um painel com aspecto de novo estimulando positivamente o técnico de manutenção.

6.7 Rede de Relacionamento

Quanto maior a rede de profissionais da mesma área, maior será a chance de você resolver algum problema específico de manutenção ou avanço tecnológico, desta forma é aconselhável que você participe de associações e grupo de estudo de sua área.

A atualização do profissional é fundamental para o seu sucesso. A atualização será feita em participações em feiras técnicas, cursos, congressos, palestras, etc. Este espaço configura-se também num espaço para aumentar nossa rede de relacionamento. Nestes momentos é fundamental anotar os meios de contato dos palestrantes e participantes do evento.

Como sugestão final cito os principais eventos que você não deve perder:

- Associação de técnicos e os engenheiros;
- Semanas de tecnologias;
- Congressos técnicos;
- Seminários e Workshops;
- Feiras de produtos elétricos e máquinas;
- Palestras e cursos de atualização profissional oferecidos por empresas e instituições de ensino.

7 - Glossário

➲ A ➲

Anilha – Sistema alfanumérico para identificação de condutores. A anilha permite numerar o cabo (Ver figura 3-16).

Aplicativo – Programa desenvolvido para um determinado tipo de máquina a partir de uma linguagem de programação.

Arquivo – Conjunto de dados armazenados em computador pessoal.

Área Classificada – Instalação industrial classificada como de segurança, onde as instalações devem minimizar algumas condições físicas (aquecimento, faiscamento, etc) evitando o comprometimento do processo e da indústria. São áreas classificadas as áreas potencialmente explosivas.

ASCII – (*American Standart Code for Information Interchange*) Código de binário que representa os caracteres.

AS-i – (*Actuator Sensor Interface*) Padrão de rede Profibus em nível de atuadores e sensores.

➲ B ➲

Back-up – Processo de armazenamento de cópia de segurança dos dados.

Baud Rate – Velocidade de transferência entre dois dispositivos expressa em número de bit por segundos.

BCD – (*Binary Coded Decimal*) Codificação de números decimais em binário.

BIOS – (*Basic Input Output System*) Sistema básico de controle dos dispositivos de entrada e saída do computador pessoal.

Bit – Unidade básica de informação do sistema digital, seu valor será 0 ou 1.

Bitmap – Desenho composto de pontos (pixels).

Borne – Conector. Dispositivo que permiti unir dois ou mais condutores. Utilizados principalmente como elo de conexão entre o painel elétrico e um dispositivo de campo, tais como: motor, sensor, etc. (Ver figura 3-15)

Bps – (Bits por segundo) Velocidade de transmissão de dados.

Buffer – Em programação é uma área de memória temporária de dados. Em sistema de hardware é um reforçador de sinal elétrico.

Byte – Conjunto de 8 bits. Em um byte armazenar 2^8 variáveis, ou seja, 256.



Call – Comando para chamar uma função.

CD – (*Compact Disc*) disco ótico compacto.

CDROM – Sistema de armazenagem de dados através de discos óticos compactos.

Clock – Pulso periódico, utilizado para acionamento de dispositivos e comandos.

CLP – Controlador lógico programável.

CNC – Comando numérico computadorizado. Utilizados em máquina de fabricação mecânica (tornos e fresas) para comando automático.

COM1 – Identificação da porta de comunicação serial 1 de um computador pessoal.

COM2 – Identificação da porta de comunicação serial 2 de um computador pessoal.

Compilador – Tradutor para linguagem de máquina. Transforma um programa gerado em uma linguagem de alto nível, para a linguagem de máquina.

Configurador – Programa de definição de características operacionais de um determinador software.

CP – Controlador programável.

CPU – (*Central Processing Unit*) Unidade de processamento central, circuito responsável pelo gerenciamento de dispositivos e execução de um programa.

CRC – (*Cyclical Redundancy Check*) Identificador de segurança com redundância cíclica, para protocolo Modbus.

CTS – (*Clear To Send*) Sinal utilizado na comunicação RS232 para indicar que o equipamento está pronto para transmissão.



DB9 – Conector padrão com 9 pinos.

DCD – (*Data Carrier Detect*) Indica que o sinal foi recebido corretamente

Disco rígido – Sistema de armazenamento de dados próprio dos computadores pessoais, mesmo que HD ou Winchester.

Diretório – Porção endereçável e agrupadora de um sistema de arquivo.

Disquete – Disco flexível portátil que permite armazenar dados.

Double word – Conjunto de 32 bit consecutivos.

Download – Operação de transferência de dados entre dois equipamentos. Enviar dados do servidor a um equipamento.

DP – (*Decentralized Peripherals*) Padrão de Comunicação Profibus a nível de periféricos descentralizados.

Driver – Programa para decodificação de protocolo que realiza a comunicação entre dois equipamentos (Supervisório). Reforçador de nível de corrente elétrica (*Hardware*).

DSR – (*Data Set Ready*) Sinal que indica que computador esta pronto para a comunicação.

DTR – (*Data Terminal Ready*) Sinal que indica que o modem esta pronto para comunicação.

DVD – (*Digital Versatile Disk*) Disco versátil digital, padrão de mídia para armazenamento de dados em disco.

➲ E ➲

E2PROM – mesmo que EEPROM

EEPROM – (*Electronically Erasable Programmable Read Only Memory*) Memória não volátil, que pode ter seu conteúdo apagado via sinal elétrico.

EIA – (*Electronic Industries Alliance*) Associação de indústrias do ramo eletrônico que determina as normas para as comunicações entre equipamentos.

EPROM – (*Erasable Programmable Read Only Memory*) Memória não volátil que pode ter seu conteúdo apagado.

EQ – (*Equal*)– sinal de comparação indicando igualdade.

E/S – Módulos de entrada e saída.

Even – Paridade impar, ou seja, a quantidade de números um na mensagem de comunicação forma um número impar.

➲ F ➲

FDB – (*Function Diagram Block*) Linguagem de programação em diagramas de blocos.

Firmware – Gerenciador de tarefas interno da máquina.

FlashEPROM – Memória programável não volátil.

Flags – Sinalizadores binários auxiliares, que servem para sinalizar certos eventos de programação.

Floint point – Ver ponto flutuante.

FMS – (*Fieldbus Message Specification*) Padrão de comunicação Profibus, para comunicação entre controlador e microcomputador.

Freeware – Forma de licença de software aberta, que não exige o pagamento de direitos autorais.

Frequência – Número de vezes que um sinal repete, ciclos idênticos, em um segundo.

Full duplex – Comunicação simultânea de recepção e transmissão.

➲ G ➲

Gateway – Ponte de comunicação entre dois sistemas.

GE – (*Grant Equal*) – Sinal de comparação indicando que um registrador é maior ou igual ao outro

GSD – (*Geräte-Stamm-Daten*)(*General Slave Data*) Dados gerais do escravo numa rede Profibus.

GT – (*Grant Than*) – Sinal de comparação indicando que um registrador é maior que outro.

➲ H ⬅

Hardware – Parte física do sistema.

Half duplex – Comunicação de transmissão e recepção de forma não simultânea.

HD – (*Hard Disk*) Ver disco rígido.

Hipertexto – Linguagem de programação simples que consiste da utilização de texto entre marcadores padrões, sinalizando a formatação de um texto. Por exemplo: <B> e </B> colocam em negrito (*bold*) o texto entre os marcadores.

HMI – (*Human Machine interface*) Mesmo que IHM.

HSC – (*High Speed Counter*) Contador de alta velocidade.

HTML – (*Hyper Text Make-up Language*) Linguagem de hipertextos utilizada em internet.

➲ I ⬅

IHM – Interface Humano Máquina / Interface Homem Máquina.

Intranet - Rede de computadores corporativa no padrão internet.

I/O - (*Input/Output*) Módulos de entrada e saída .

➲ J ⬅

Jump – Comando para saltar uma parte de um programa.

➲ L ➲

LAN – (*Local Area Network*) Rede de computadores local.

Laptop – Computador pessoal de uso portátil.

LCD – (*Liquid Crystal Display*) Tipo de display que usa cristal liquido entre dois painéis de eletrodos transparente

LE – (*Less Equal*) – Sinal de comparação que indica que um registrador é menor ou igual a outro registrador.

LED – (*Light Emitter Diode*) Semicondutor que emite luz na presença de corrente elétrica.

LT – (*Less Than*) – Sinal de comparação que indica um registrador é menor que outro.

➲ M ➲

Master-Slave – Rede de comunicação no padrão mestre-escravo

Mídia – Meio físico para armazenamento e apresentação de informação

Memória retentiva – Memória que mantém seu conteúdo intacto após uma desenergização do sistema.

Menu – Lista de opções de uma tela.

Microcontrolador – Circuito integrado que possui em um mesmo invólucro: microprocessador, memória e portas de comunicação.

Microprocessador – Circuito integrado que executa instruções e monitora o controle de dispositivos.

MMI – (*Man Machine Interface*) Mesmo que IHM.

➲ N ➲

Navram – Memória RAM especial, com alimentação por bateria externa, que mantém o conteúdo armazenado mesmo com o equipamento desligado.

NEMA – (*National Electrical Manufacturers Association*) Associação norte-americana que define padrões elétricos.

➲ O ➲

Odd – Paridade par. Informa se a quantidade de números um numa mensagem de comunicação é um número par.

Overflow – Erro de comunicação serial que se caracteriza pela recepção de muitos dados, não permitindo a identificação dos dados pelo receptor.

Overram – Erro de comunicação serial que se caracteriza pela falta de dados na comunicação.

➲ P ➲

PA – (*Process Automation*) Padrão de comunicação Profibus a nível de automação de processo.

Paridade – Método de informação utilizado em comunicação serial para identificar a validade de transmissão. Consiste em transmitir um caractere extra a fim de determinar se o somatório de números “1” transmitidos na mensagem é par (*odd*) ou impar (*even*).

PC – (*Personal Computer*) Computador pessoal.

Pendrive – Dispositivo de armazenamento de dados com Flasheprom para conexão USB.

PLC – (*Programmable Logic Controller*) Controlador lógico.

Ponto flutuante – Representação de um valor numérico através de uma matriz e um multiplicador na base 10.

Posta-em-marcha – Processo de ativação em chão-de-fábrica de um sistema automático, com testes, correções e entrega do sistema

Potencialmente Explosiva – Áreas industriais com gases em suspensão. Ver também área classificada.

Profinet – (*Profibus for Ethernet*) Padrão de comunicação Profibus a nível de rede gerencial.

PROM – (*Programmable Read Only Memory*) Memória não volátil que programada uma única vez.

Protocolo de Comunicação – Conjuntos de regras que define a troca de informações entre dois ou mais equipamentos.

PV – (*Process Variable ou Present Value*) Variável de processo corrente.

PWM – (*Pulse Width Modulation*) Modulação por largura de pulso.



Query – Consulta de dados



RAM – (*Random Access Memory*) Memória volátil de acesso aleatório para leitura e escrita.

Reset – Inicialização de um sistema ou variável.

Retry – Retransmissão de informações.

Return – Comando para retorno de uma função.

RI – (*Ring Indicator*) Indicador de chegada de sinal para comunicação.

ROM – (*Read Only memory*) Memória não volátil de apenas leitura.

Rotulo – Identificação alvo para um comando de salto ou função, mesmo que label.

RS232 – Padrão de comunicação (Ver capítulo 6).

RS422 – Padrão de comunicação (Ver capítulo 6).

RS485 – Padrão de comunicação (Ver capítulo 6).

RTC – (*Real Time Clock*) Relógio em Tempo Real.

RTS – (*Request To Send*) Sinal utilizado em comunicação RS232 indicando que o equipamento está pronto para receber dados.

RTU – (*Remote Terminal Unit*) Padrão de configuração do protocolo Modbus.

Rx – (*Received Data*) Sinal de recepção serial de dados
RxD – Mesmo que Rx.

➲ S ➲

SCAN –Tempo de varredura e execução de um programa

SCADA – (*Supervisory Control and Data Acquisition*) Sistema de supervisório com aquisição de dados.

Script – Pequeno programa utilizado na configuração de supervisórios que são ativados com algum evento relacionado a um objeto.

Serial – Técnica de comunicação que utiliza um par de fios para transmitir os dados bit a bit sequencialmente.

SFC – (*Sequential Function Chart*) Programação com a utilização de sequências de função.

Shareware – Forma de licença de software semi-aberta, que exige para funcionamento um registro gratuito, sendo o pagamento de direitos autorais feito através de doação compulsória.

Simplex – Comunicação apenas de transmissão ou de recepção.

SP - (*Set Point*) Variável de Referência para controle.

Software – Conjuntos de atividades pré-estabelecidas para um sistema automático, mesmo que programa.

Start-up – o mesmo que posta-em-marcha.

Stop bit – Em uma comunicação assíncrona, um ou mais bits que indicam o fim de um bloco de caracteres.

Subrotina – Parte de um programa reservado e que realiza funções específicas, que chamado e executado de várias partes do programa principal.

SSR – (*State Solid Relay*) Rele de estado sólido.

SV – (*Set Value*) Variável de referência para controle.

➲ T ➲

Tag – Para comentários de lógicas: Identificador de um operando de controle.

Tag – Para supervisórios: Variável alfanumérica envolvida no processo.

Time-out – Intervalo de tempo tolerado para a troca de informações.
Toggle – Troca, botão liga e desliga
Trend – Análise gráfica.
Tx – (Transmitted Data) Sinal de transmissão de um cabo serial.
TxD – Mesmo que Tx.
TXT – Arquivo de programa com formatação em texto ASCII.

➲ U ➲

UCP – (Unidade Central de Processamento) Mesmo que CPU.
Upload – Enviar dados para servidor. Ler dados de um equipamento e armazenar no servidor.
USB – (Universal Serial Bus) Padrão de comunicação serial para conexão de equipamentos.

➲ W ➲

Watch Dog – (Cão de Guarda) Sistema de monitoração de falhas de uma CPU, que verifica um sinal com base de tempo constante gerado pelo software. Na falta deste sinal a CPU gera o alarme de watch-dog que bloqueará o processamento e as saídas do controlador.
Winchester – ver disco rígido
Wire-less – (Sem fio) Sistema de comunicação, sem fios, através de ondas eletromagnéticas.
Word – Conjunto de 16 bit ou 2 bytes.

➲ X ➲

XML – (*eXtend Make-up Language*) Linguagem HTML com código ampliado

8 - Referências

Bonacorso, Nelso Gauze; Noll, Valdir, **Automação Eletropeneumática**, São Paulo: Érica, 1997.

Carvalho, Cláudio Sergio da Rós de, **Microprocessador 8085**, Campinas: Editora da Unicamp, 1988.

Capelli, Alexandre, **Automação Industrial: Controle do Movimento e Processos Contínuos**, São Paulo, Editora Érica, 2008

Cassiolato, César; Torres, Leandro H.B.; Camargo, Paulo Roberto, **Profibus – Descrição Técnica**, Associação Profibus, São Paulo, 2006

Catálogo Geral de Segurança Intrínseca, Sense Sensores e Instrumentos, São Paulo, 2002.

Catálogo Instrumentação para Medição e Controle, Comtemp, São Caetano do Sul: Comtemp, 2003.

Catálogo Linha de Produtos, Poliron Cabos Elétricos Especiais Ltda, Diadema: Poliron, [2002]

Catálogo Máster 2001, LR Informática Industrial, Porto Alegre: LR, 2001.

Catálogo Photoelectric Sensors, Banner Engineering Corporation, Minneapolis-USA, 2001.

Catálogo Sensores Abraf Sensor, Abraf Sensor Controles Eletrônicos Ltda, São Paulo, [2000].

Controlador Programável MPC4004, Atos Automação Industrial Ltda, São Paulo: Atos, 2001.

Fialho, Arivelto Bustamante, **Instrumentação Industrial: Conceitos, Aplicações e Análises**, São Paulo: Érica, 2007

Gaidzinski, Vladimir Hartenias, **Automação I**, apostila curso: Tecnólogo em Automação Industrial, SATC: Criciúma, 2001.

Georgini, Marcelo, **Automação Aplicada, Descrição e Implementação de Sistemas seqüenciais com PLCs**, São Paulo: Érica, 2000.

Leite, Duílio Moreira, **Esquemateca: Tecnologia do Controle e Automação Industrial**, São Paulo: MM Editora, 2000.

Lima, Walter Celso, **Dispositivos Sensores de Erro**, Florianópolis: Editora da UFSC, 1984.

Manuale dell'Operatore PLC Sysmac CQM1, Omrom, Itália: Omron, 1995.

Manual de Comandos de Pesagem para Modbus RTU/ASCII, Alfa Instrumentos, Revisão 2.0, São Paulo: Alfa, 2002.

Manual de Instalação e Programação do Controlador Programável TP02, WEG Automação, Jaraguá do Sul: Weg, 2001.

Manual de Instalação e Programação do Controlador Programável TPW03, WEG Automação, Jaraguá do Sul: Weg, 2007.

Manual de Programação do Controlador BCM, BCM Automação, Porto Alegre: BCM, [1997].

Manual de programação e operação FX2N, Mitshubshi, Japão: Mitshubshi, 1999.

Manual de Programação Hitachi, São Paulo: Hitachi ,1997.

Manual de Programação Máster Tool, Altus Sistemas de Automação, Porto Alegre: Altus, 1998.

Manual Profibus-DP, Saia PCD – Process Control Devices, English edition 26-765 E1, SAIA-Burgess Electronics Ltd., Switzerland, 1999.

Micro-controlador Programável TP01-TP11, WEG Automação,Jaraguá do Sul: Weg, 1997.

Mielle, Fabio, **Softwares de Controle Evoluem e Ganham mesma Importância que Hardware**, Artigo Revista Controle & Instrumentação, pág. 90 e 91, São Paulo, Março de 2003.

Norton, Peter, **Introdução à informática**, São Paulo: Makron Books, 1996.

Oliveira, Júlio César Peixoto de, **Controlador Programável**, São Paulo: Makron Books, 1993.

Programando com AL3830, Altus Sistemas de Automação, Porto Alegre: Altus, 1990.

Programmable Controller FP0 – Programming Manual, Nais – Matsushita Automation Control, Japão: Nais,1999.

Programmable Controller FP0 – Hardware, Nais – Matsushita Automation Control, Japão: Nais, 1999.

Pupo, Mauricio Santos e Gonzaga, Adilson, **Supervisão de um CLP em Controle de Processos através da WWW**, Artigo Revista Controle & Instrumentação, pág. 78 a 83, São Paulo, Março de 2003.

Santos, José J. Horta, **Automação Industrial: Uma Introdução**, Rio de Janeiro: Livros Técnicos e Científicos, 1979.

Santos, E.W. E Silveira, P.R., **Automação e Controle Discreto**, São Paulo: Érica, 2002.

Simatic S7 Programmable Controller, Siemens AG, Alemanha: Siemens, 1998.

Souza, Giovani Batista de, **Automação I**, apostila curso: Tecnólogo em Automação Industrial, SATC: Criciúma, 2006.

Souza, Giovani Batista de, **Automação II**, apostila curso: Tecnólogo em Automação Industrial, SATC: Criciúma, 2007

Souza, Giovani Batista de, **Controladores Lógicos**, apostila curso: Tecnólogo em Eletromecânica, SATC: Criciúma, 2008

SYSMAC CQM1H Series : Operation Manual, Japão: Omron, [2005].

Walisiewicz, Marek, **Energia Alternativa: Solar, Eólica, Hidroelétrica e de Biocombustíveis**, São Paulo: Publifolha, 2008.

9 - Apêndice I - Caderno Exercício - Siemens

Neste apêndice acompanhamos os exercícios do livro para a configuração dos controladores série 200 da Siemens.

9.1 Endereçamento

Na série 200 da Siemens os endereços dos operandos de memória e dispositivos são feitos com base em blocos de 1 byte. O diagrama da figura 12-1 corresponde ao endereçamento dos operandos desta série.

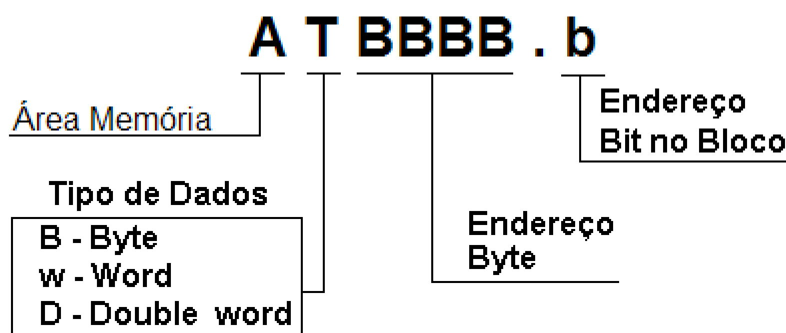


Figura 12-123 Endereçamento Siemens

Todos os endereços são acessados em blocos de 8, 16 ou 32 bits. Seu endereço é dado através do nome do bloco seguido de B, W ou D para 8, 16 ou 32 bits respectivamente, mais o endereço do primeiro byte de memória correspondente. Para endereçar um bit individualmente, informar o nome do bloco juntamente com o endereço do byte, separando o número do bit com um ponto.

Descrição	Endereço	Tipo	Rang e	Endereço	Tipo	Range
Input Register	IB0 a IB7	Byte	-128 a 127	I0.0 a I7.7	Binário	0 a 1
Entradas	Manipulação do bloco entrada			Manipulação dos endereços individuais		

Descrição	Endereço	Tipo	Rang e	Endereço	Tipo	Range
Output Register	QB0 a QB7	Byte	-128 a 127	Q0.0 a Q7.7	Binário	0 a 1
Saídas	Manipulação do bloco de saídas			Manipulação dos endereços individuais		

Descrição	Endereço	Tipo	Rang e	Endereço	Tipo	Range
Memory Variable	MB0 a MB31 e VB0 a VB5119	Byte	-128 a 127	M0.0 a M31.7 e V0.0 a V5119.7	Binário	0 a 1
Memória Auxiliar	Manipulação de bloco de 8 bits			Manipulação dos endereços individuais		

Descrição	Endereço	Tipo	Range	Endereço	Tipo	Range
Variable	Mw0 a Mw30 VW0 VW5118	Word	-32768 32767	MD0 a MD28 VD0 a VD5116	Double Word	**-2^{32} a** **$2^{32}-1$**
Memória Auxiliar	Manipulação de bloco de 16 bits			Manipulação de bloco de 32 bits		

Descrição	Endereço	Tipo	Range	Endereço	Tipo	Range
Special Memory	SMB0 a SMB194	Byte	0 a 255	SM0.0 a SM194.7	Binário	0 a 1
Memória Sistema				Pontos binários interno do sistema. P.Ex. SM0.5 – Pulso de um segundo		

Descrição	Endereço	Tipo	Range	Endereço	Tipo	Range
Timer retentive on delay	T0 e T64	Word	O a 65535	T0 e T64	Binário	0 a 1
Temporizador	Acumulador dos temporizadores em milésimos de segundos. O range corresponde a zero a 65,535 segundos.			Informa o estado do temporizador Utilizado pela instrução de temporização com reset.		

Descrição	Endereço	Tipo	Range	Endereço	Tipo	Range
Timer retentive on delay	T1 a t4 e T65 a T68	Word	O a 65535	T1 a t4 e T65 a T68	Binário	0 a 1
Temporizador	Acumulador dos temporizadores em centésimos de segundos. O range corresponde a zero a 655,35 segundos.			Informa o estado do temporizador Utilizado pela instrução de temporização com reset.		

Descrição	Endereço	Tipo	Range	Endereço	Tipo	Range
Timer retentive	T5 a T31	Word	O a 65535	T5 a T31 e	Binário	0 a 1

on delay	e T69 a t95			T69 a t95		
Temporizador	Acumulador dos temporizadores em décimos de segundos. O range corresponde à utilização de temporizadores com até 6553,5 segundos			Informa o estado do temporizador Utilizado pela instrução de temporização com reset.		

Descrição	Endereço	Tipo	Range	Endereço	Tipo	Range
Timer on/off delay	T32 e T96	Word	O a 65535	T32 e T96	Binário	0 a 1
Temporizador	Acumulador dos temporizadores em milésimos de segundos, permitindo temporização máxima de 65,535 segundos.			Informa o estado do temporizador		

Descrição	Endereço	Tipo	Range	Endereço	Tipo	Range
Timer on/off delay	T33 a t36 e T97 a T100	Word	O a 65535	T33 a t36 e T97 a T100	Binário	0 a 1
Temporizador	Acumulador dos temporizadores em centésimos de segundos. O range máximo fica em 655,35 segundos.			Informa o estado do temporizador.		

Descrição	Endereço	Tipo	Rang e	Endereço	Tipo	Range
Timer on/off delay	T37 a T63 e T101 a t255	Word	O a 65535	T37 a T63 e T101 a t255	Binário	0 a 1

Temporizador	Acumulador dos temporizadores em décimos de segundos, permitindo tempo de até 6553,5 segundos.	Informa o estado do temporizador

Descrição	Endereço	Tipo	Rang e	Endereço	Tipo	Range
Counter	C0 a C255	Word	O a 65535	C0 a C255	Binário	0 a 1
Contador	Acumulador dos Contadores			Estado do contador		

Descrição	Endereço	Tipo	Rang e	Endereço	Tipo	Range
Analog Input	AIW0 a AIW30	Word	O a 65535			
Entrada analógica	Leitura dos dados analógicos					

Descrição	Endereço	Tipo	Rang e	Endereço	Tipo	Range
Analog Output	AQW0 a AQW30	Word	O a 65535			
Saída analógica	Controle placa analógicos					

Descrição	Endereço	Tipo	Range	Endereço	Tipo	Range
High speed counter	HC0 a HC5	Doubl e	0 a 2^{32}-1			
Contador Alta Velocidade	Contadores de Alta velocidade					

Tabela 12-12 Endereçamentos operandos CP Siemens

As constantes são representadas em qualquer base e em ASCII. Por exemplo: uma word com o número 10 em hexadecimal será representada por: 16#000A.

9.2 Software de Programação e Funções

O software de programação utilizado é o Step 7 – MicroWin 32.

No software de programação Step 7 – MicroWin 32, selecionar os seguintes comandos;

- Contato NA e NF
- Rele Bobina
- Temporizador
- Contator
- Ligações verticais e horizontais
- Funções Set e Reset
- Contatos de transição
- Funções especiais

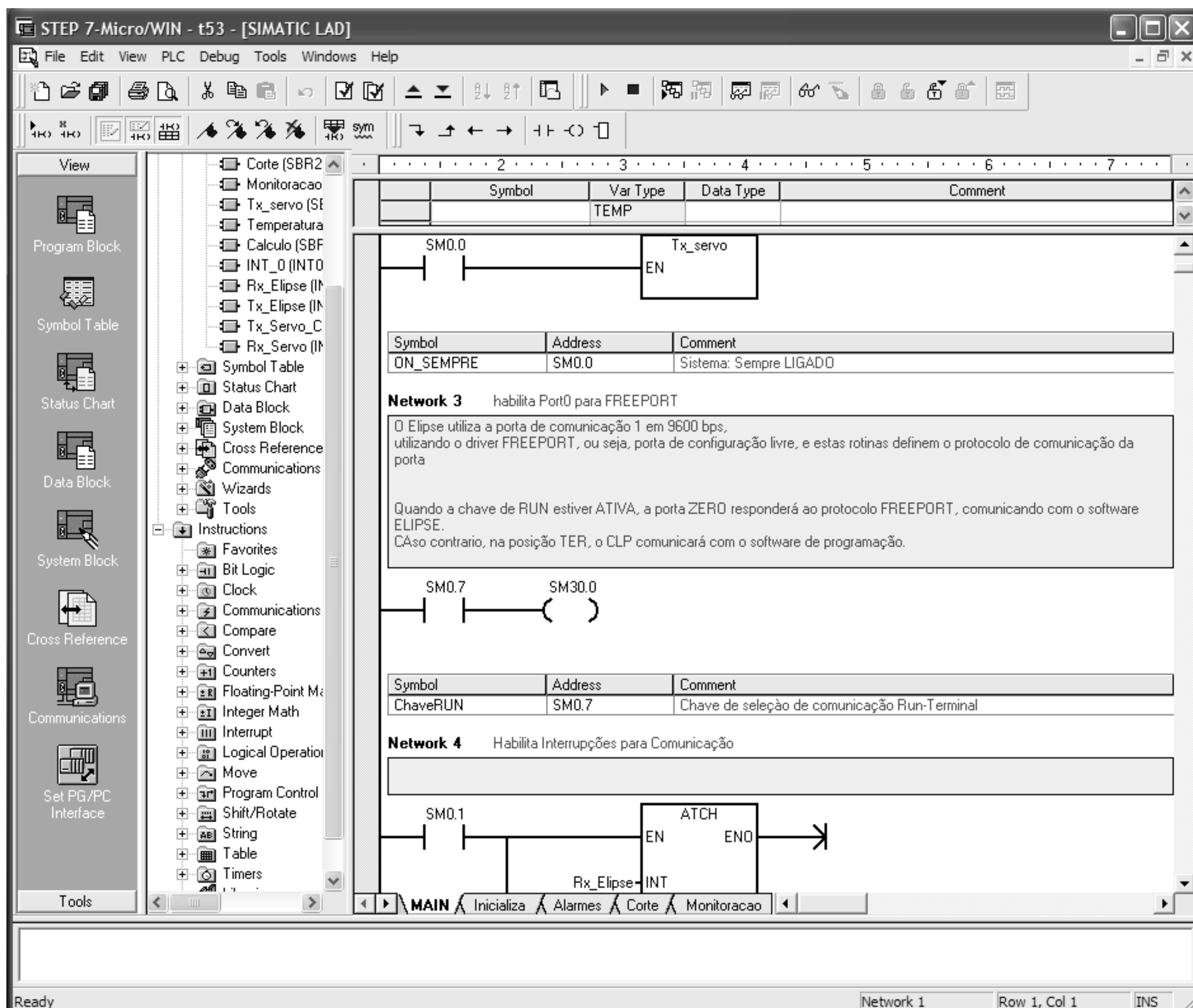


Figura 12-124 Software de programação Siemens

No help do software são encontradas com detalhamento as funcionalidades de cada item utilizado para a programação do CP série 200. A maioria das funções que foram listadas neste livro tem correspondência com as funções utilizadas no configurador Step 7 – MicroWin 32. No software de programação estas funções são agrupadas conforme sua utilização, como segue:

- Bit Logic (Contatos e bobinas)
- Clock (RTC)

- Communications (Comunicação serial)
- Compare (Comparação)
- Convert (Conversação)
- Counters (Contadores)
- Floating-Point Math (Manipulação de número com pontos flutuante)
- Integer Math (Operações aritméticas inteiras)
- Interrupt (Interrupções)
- Logical Operation (Operações booleanas)
- Move (Movimentação de dados)
- Program Control (Saltos e Subrotinas)
- Shift-Rotate (Deslocamento de registradores)
- String (Manipulação de ASCII)
- Table (Manipulação de tabelas)
- Timers (Temporizadores)

9.3 Configuração

Não existem configurações para os programas básicos feitos em ladder para o Step 7 – MicroWin 32. Somente será necessário a configuração quando se utiliza à comunicação serial, a utilização de IHM, contador de alta velocidade e nestes casos utilizamos as memória do sistema < SM > para realizar esta configuração via ladder.

9.4 Inicializando a Comunicação

No software S7_200 - MicroWin para inicializar a configuração com o controlador da série 200, basta solicitar através do comando a operação < UPLOAD >, < DOWNLOAD > e ou < monitoração >.

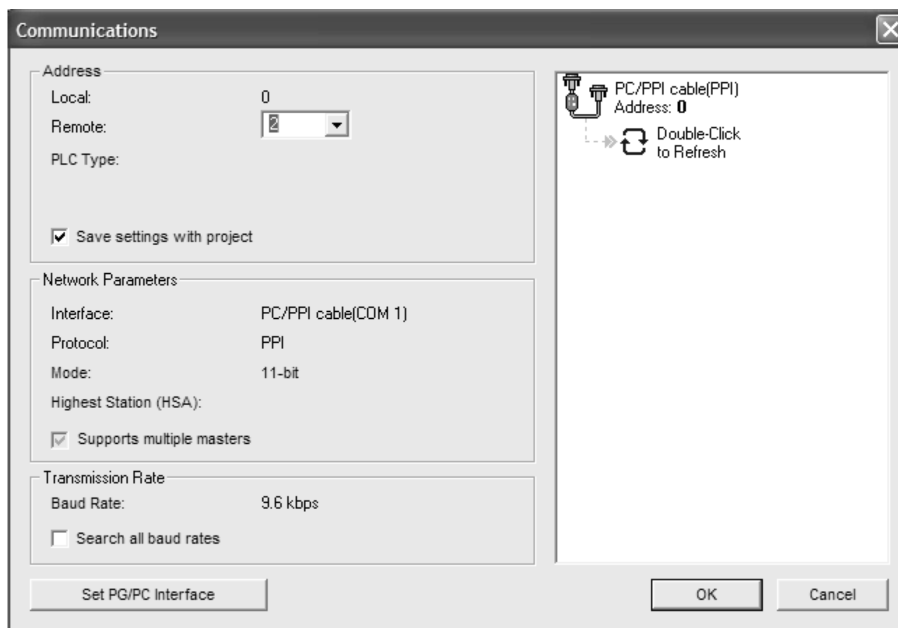


Figura 12-125 Configuração canal serial Siemens

Para configurar o canal serial assim como o número da porta de comunicação e o endereço do controlador na rede, utiliza-se a opção de < UPLOAD > ou < DOWNLOAD > e selecionando-se < COMMUNICATIONS >. A figura 12-3 corresponde à tela que será apresentada para configuração da comunicação.

Para transferir um programa é necessário que o controlador esteja em modo < STOP >.

9.5 Primeiros Diagnósticos

O funcionamento do controlador é indicado num primeiro momento pelo estado dos LEDs de sinalização do controlador. A tabela 12-2 mostra o estado dos LEDs conforme o funcionamento do CP.

Estado de Funcionamento	**RUN (Verde)**	**STOP (Laranja)**	**SF (Vermelho)**
MODO RUN – Monitoração e modo de alteração	L	D	D
MODO STOP – Modo de Programação	D	L	D
MODO ERRO	D	L	L
Erro de bateria – Baixa tensão bateria	L	D	L
Sem programa	D	L	L

Erro de alimentação	D	D	D

Tabela 12-13 Estado leds do CP Siemens

Ficha de Avaliação

Nossa instituição avalia os alunos por competências atribuindo as seguintes notas:

- E = Excelente
- P = Proficiente
- S = Suficiente
- I = Insuficiente

A tabela a seguir deve ser utilizada por você para ajudá-lo a acompanhar o processo de ensino e aprendizagem na disciplina de automação. Nesta tabela nas colunas são apresentadas as habilidades e atitudes que serão avaliadas no transcorrer do semestre. Você deve ir preenchendo com os conceitos que lhe serão atribuídos nas avaliações do semestre.

	Conhece o CLP	Reconhece a importância da Automação	Conhece dispositivos de entrada e saida	Conhece programação LADDER	Descreve as formas de controle e comnado com CLP	Descreve a finalidade e identificação da estrutura do CLP	Diferencia módulos de entra/saida do CLP	Descreve o funcionamento e a montagem de dispositivos	Interage com a máquina e utiliza dispositivos de comunicação	Aplica e utiliza adequadamente os comandos de programação	Desenvolve programas adequados para atender as necessidades apresentadas		